Departamento de Electrónica Física
Escuela Técnica Superior de Ingenieros de Telecomunicación

Artificial recurrent neural networks for the distributed control of electrical grids with photovoltaic electricity

Autor:

Eduardo Matallanas de Ávila
Ingeniero de Telecomunicación

Directores:

Estefanía Caamaño Martín
Dra. Ingeniera de Telecomunicación

Álvaro Gutiérrez Martín
Dr. Ingeniero de Telecomunicación

2016

Tribunal nombrado por el Mgfco. y Excmo. Sr. Rector de la Universidad Politécnica de Madrid.

Presidente/a:	Dr. D. Félix Monasterio-Huelin Maciá
Secretario/a:	Dr. D. Miguel A. Egido Aguilera
Vocal 1:	Dr. D. Leocadio Hontoria García
Vocal 2:	Dr. D. Iñaki Navarro Oiza
Vocal 3:	Dr. D. Anders Lyhne Christensen
Suplente 1:	Dr. D. Pablo Díaz Villar
Suplente 2:	Dr. D. Manuel Castillo Cagigal

Realizando el acto de defensa y lectura de la Tesis en la E.T.S.I.T., Madrid,

el día ________ de ____________ de 2016.

Acuerdan otorgar la calificación de: ________________________________

Y, para que conste, se extiende firmada por los componentes del tribunal, la presente diligencia

El/La Presidente/a El/La Secretario/a

Fdo: ______________ Fdo: ______________

El/La Vocal 1 El/La Vocal 2 El/La Vocal 3

Fdo: ______________ Fdo: ______________ Fdo: ______________

Resumen

El sistema eléctrico actual no ha evolucionado desde sus orígenes. Esto ha hecho que emerjan diferentes problemas los cuales son necesarios afrontar para incrementar el rendimiento de las redes eléctricas. Uno de estos problemas es el crecimiento de la demanda, mientras que otros, como las Tecnologías de la Información y las Comunicaciones (TIC) o la Generación Distribuida (GD), se han desarrollado dentro de la red eléctrica recientemente sin ser propiamente integradas dentro de ella. Esta Tesis afronta los problemas derivados del manejo y la operación de las redes eléctricas existentes y su evolución hacia lo que se considera la red eléctrica del futuro o Smart Grid (SG).

El SG nace de la convergencia de cinco aspectos: i) la red eléctrica, ii) TICs, iii) energías renovables, iv) almacenamiento eléctrico y v) Gestión de la Demanda Eléctrica (GDE). Esta Tesis consiste en un primer paso hacia el SG uniendo e integrando los cinco aspectos claves para su desarrollo y despliegue en el futuro cercano. Para ello, la mejora del estado de la red eléctrica se consigue a través del suavizado del consumo agregado. Para lograr este objetivo, se propone el uso de un algoritmo que procese la información proveniente de las TICs para que todas las partes de la red eléctrica se puedan beneficiar. Algunos de estos beneficios son: mejor uso de las infraestructuras, reducción de su tamaño, mayor eficiencia, reducción de costes e integración de GD, entre otros. El algoritmo propuesto está basado en una aproximación distribuida en la que los usuarios son hechos partícipes de sus decisiones, siendo capaces de manejar sus flujos de potencia con este objetivo. El algoritmo se ha implementado siguiendo una estrategia basada en la GDE combinada con el control automático de la demanda que ayude a integrar los Recursos Energéticos Distribuidos (RED) (energías renovables y sistemas de almacenamiento eléctrico), que lo conducen hacia un concepto innovador denominado Gestión de la Activa de la Demanda Eléctrica (GADE).

En esta Tesis, una aproximación basada en la Inteligencia Artificial (IA) ha sido utilizada para implementar el algoritmo propuesto. Este algoritmo ha sido construido utilizando Redes Neuronales Artificiales (RNAs), más concretamente Redes Neuronales Recurrentes (RNRs). El uso de RNAs ha sido motivado por las ventajas de trabajar con sistemas distribuidos, adaptativos y no lineales. Y la elección de las RNRs se ha basado en sus propiedades dinámicas, las cuales encajan perfectamente con el comportamiento dinámico no lineal de la red eléctrica. Además, un controlador neural es utilizado para manejar cada elemento de la red eléctrica, incrementado la eficiencia global a través del suavizado del consumo agregado y maximizando el autoconsumo de los RED disponibles. Sin embargo, no existe ningún tipo de comunicación entre los distintos individuos y la única información disponible es el consumo agregado de la red eléctrica. Finalmente, la mejora de la red eléctrica se ha conseguido de manera colectiva utilizando el algoritmo propuesto para coordinar la respuesta del conjunto de controladores neuronales.

Abstract

The present electrical systems have not evolved since its inception. This fact has triggered the emergence of different problems which are necessary to tackle in order to enhance the grid performance. The demand growth is one of these problems, while others, such as the Information and Communications Technology (ICT) or Distributed Generation (DG), have been recently developed inside the grid without their proper integration. This Thesis addresses the problems arising from the management and operation of existing electrical grids and their evolution to what is considered the grid of the future or Smart Grid (SG).

The SG is born from the convergence of five aspects: i) the grid, ii) ICTs, iii) renewable energies, iv) Electrical Energy Storages (EESs) and v) Demand Side Management (DSM). This Thesis consists of a first step towards the SG by linking and integrating the five key aspects for its development and deployment in the near future. To this end, the enhancement of the grid status is achieved by the smoothness of the aggregated consumption. In order to fulfill this objective, an algorithm has been proposed that processes the data gathered from the ICTs to benefit all the parts of the grid. Some of these benefits are: better use of the infrastructure, reduction in its size, greater operational efficiency, cost reductions and integration of the DG, among others. The proposed algorithm is based on a decentralized approximation in which the users are made participants in their decisions, being able to manage their power flows into this objective. It is implemented following DSM techniques combined with an automatic control of demand that helps to integrate Distributed Energy Resources (DER) (renewable energies and EESs), which leads to an innovative concept called Active Demand Side Management (ADSM).

In this Thesis, an Artificial Intelligence (AI) approach was used to implement the proposed algorithm. This algorithm is built based on Artificial Neural Networks (ANNs), specifically Recurrent Neural Networks (RNNs). The use of ANNs is motivated by the advantages of working with distributed, adaptive and nonlinear systems. And the election of RNNs is based on their dynamic behavior, which fits perfectly with the nonlinear dynamic behavior of the grid. In addition, a neural controller is used to operate in each element of the grid to increase the global efficiency by smoothing the aggregated consumption and to maximize the local self-consumption of the available DER. However, there is no communication among the users and the available information is only the aggregated consumption of the grid. Finally, the enhancement of the grid is achieved collectively by using the proposed algorithm to coordinate the responses of the neural controller ensemble.

To my Family

Acknowledgements

After all these years, it is extremely difficult to condense all the thanks to all those involved in the development of the Thesis. But I have already finished my Ph.D. without dying in the process, so I think I will be able to do it.

First of all, I am truly grateful to my supervisors Prof. Alvaro Gutiérrez Martín and Prof. Estefanía Caamaño Martín for the opportunity they gave me to take this path. Alvaro is like my science father while Estefanía is my science mother. Alvaro is the culprit that started this madness and I really thank his effort and constant support to the development of this Thesis. He always had time to discuss and search for a solution to all the problems found during the development of the Thesis. He has also put up with me during these years. For all his support and help, my deepest gratitude Alvaro. On the other hand, I would like to thank Estefanía all her concerns and interests in the work done during the development of this Thesis. She was very supportive and thorough review of the work done has been a source of valuable improvements.

I am grateful to have been part of the ROBOLABO family during the realization of this Thesis. Here I have met many people: teachers, students and laboratory teachers who have made the development of this Thesis possible. My fellow sufferer Dr. Manuel Castillo Cagigal has shared with me the entire path until the end of the Thesis. Our discussions, laughs, concerns and everything was of great help to survive this Thesis. Without his support, I would not have made it till the end. I would also express my deepest gratitude towards Dr. Iñaki Navarro Oiza. He was the one that helped me with the organization of the Thesis, and always supported me to continue with it in spite of all the problems. He was always there to support, discuss and direct my efforts towards the fulfillment of this Thesis. My special thanks to Marina Calvo, she was in the lab for a couple of years but she became a close friend and we helped each other during her stay in the lab. Another good friend and roomy is Miguel Gómez Irusta, he was in the lab for a few months but he became a close friend in my life. He was always there especially for the worst moments of the Thesis and his support has been essential to end this Thesis. During the end of the Thesis, I was able to meet Juana González-Bueno Puyal, María Peral Boiza, Raquel Higón and Marcela Calpa. They were very supportive and their encouragement helps me in the boring and difficult moments. Specially, Juana with her smile, her positive view of life and the beers, do not ever change. And María always making fun of me but always worried about me and helping with my mental condition. I am grateful to have meet Prof. Félix Monasterio-Huelin. He was the one who changed my mind while still being an undergraduate and his way of teaching made me rethink all I knew. I was able to discuss with him part of this Thesis and he helped me in the development of my research skills. In spite of the constant hammering, I would also like to thank Rafael Rodríguez Merino, the laboratory teacher, for his support.

I wish to thank Prof. Carolina Sánchez Urdaín, Prof. Jesús Fraile Ardanuy and Prof. Benito Artaloytia Encinas for their support and concerns around the Thesis. I would also like to thank them to give me the opportunity to collaborate with them in the development of their energy subject.

I am also grateful to have been part of GEDIRCI during the realization of my Thesis. I want to thank Dr. Daniel Masa, Dr. Jorge Solórzano, Dr. Lorenzo Olivieri and Prof. Miguel Angel Egido their support and advice in the energy field. I would also like to mention my thanks to my colleagues of the Solar Energy Institute who have been with me during the development of this Thesis: Dr. Iñigo Ramiro, Dr. Rodrigo Moretón and Ana Peral Boiza between others.

I wish to thank Prof. Bryan Tripp for the opportunity to do a research internship in the Neuro Robotics lab of University of Waterloo. He taught me all I know about computational neuroscience. His advice helped me to center the efforts of this Thesis and to improve as researcher. I wish to thank the people who also helped me in the development of my work in Waterloo: Omid Rezai, Ben Selby and Ashley Kleinhans between others.

I would also like to thank the international reviewer Dr. Alexandre Campo and Dr. Lucia Gauchia for reviewing the document and providing useful comments for its improvement.

I thank the sponsorship of the Spanish Ministry of Education with a Ph.D. Grant (FPU-2011) for the entire period of realization of this Thesis. I am grateful to GeDELOS-FV and SIGMAPLANTAS projects by the possibility of using the necessary equipment for the development of this Thesis. I also wish to thank the Escuela Técnica Superior de Ingenieros de Telecomunicación of the Technical University of Madrid for their support in the maintenance and use of "MagicBox" prototype.

I wanted to thank my friends, Rober, Adri, Rafa, Guille, Emilio, Víctor, Pablo, Alberto and Juan between others, for their support and for always being there in spite of changing the parties for books or papers. Thank you guys for all the good moments that we have had during these years and all the laughs. In addition, I wanted to thank all the "Comando Cimero" crew, being part of lots of laughs and good moments during this time. All of them were always concerned about the Thesis and my condition during these years.

I do not know how to express all the thankfulness I feel towards my family. Starting with my girl, Martha, who has suffered this Thesis as it was her own. She has been a solid rock to me during these years. Her understanding and patience during the hard times that arise in the development of a Ph.D. Thesis were essential to achieve the end of the path. This Thesis is also part of her. I would also want to thank my brothers, Javi and Jorge, to be very supportive during this period of my life despite being a pain in the ass. Special thanks to my parents, Paco and Libo, without them continuous studying would never be possible, they gave everything that I needed and their uncoditional love helps to end this Thesis. Thanks to them I am now writing these acknowledgements, and it is something that can never be repaid. I would also like to thank Martha's family that during these years has become also my family. I wanted to thank them to be so supportive and concerned about the development of this Thesis.

Finally, to all of you, who are not explicitly named and who have been part of the development of this Ph.D. Thesis

Thank you!

Contents

List of Figures

List of Tables

List of Algorithms

Nomenclature

τLA τ-Learning Algorithm

AC Alternating Current

ADALINE ADAptative LINear Elements

ADSM Active Demand Side Management

AI Artificial Intelligence

AMI Advanced Metering Infrastructure

ANC Active Noise Control

ANN Artificial Neural Network

ART Adaptive Resonance Theory

BDA Big Data Analytics

BIBO Bounded Input Bounded Output

BPL Broadband over Power Line

BSS Blind Source Separation

CPP Critical Peak Pricing

CTRNN Continuous Time RNN

CX Cycle crossover

DC Direct Current

DER Distributed Energy Resources

DG Distributed Generation

DL Deep Learning

DR Demand Response

DSM Demand Side Management

DTRNN Discrete Time RNN

EDeNC Evo-Devo Neural Controller

EES Electrical Energy Storage

EHV Extra High Voltage

ENTSO-E European Network of Transmission System Operators

ESN Echo State Network

EU European Union

EV Electric Vehicle

FPGA Field Programmable Gate Array

GA Genetic Algorithm

GPU Graphics Processing Unit

HAN Home Access Network

HEV Hybrid Electric Vehicle

HV High Voltage

HVAC Heating, Ventilation and Air Conditioning

ICT Information and Communications Technology

IEA International Energy Agency

IoT Internet of Things

IQR InterQuartile Range

LAN Local Area Networks

LMS Least Mean Squares

LSTM Long Short Term Memory

LV Low Voltage

ML Machine Learning

MV Medium Voltage

NAN Neighborhood Area Network

NARMAX Nonlinear Autoregressive Moving Average with eXogenous inputs

NEF Neural Engineering Framework

NGA Neural Grid Algorithm

OX Order crossover

PCA Principal Components Analysis

PHEV Plug-in Hybrid Electric Vehicles

PLC Power Line Carrier

PMX Partially matched crossover

PTR Peak Time Rebates

PV Photovoltaics

QoS Quality-of-Service

RBF Radial Basis Function

REE Red Eléctrica de España

rms Root Mean Square

RNN Recurrent Neural Network

ROC Region of Convergence

RTP Real Time Pricing

SCADA Supervisory Control And Data Acquisition

SG Smart Grid

SI International System of Units

SI Swarm Intelligence

SNARC Stochastic Neural Analog Reinforcement Computer

SoC State of Charge

SOM Self-Organizing Map

SPAUN Semantic Pointer Architecture Unified Network

SVM Support Vector Machine

TOU Time of Use

UHV Ultra High Voltage

UPS Uninterruptible Power Supply

V2G Vehicle-to-Grid

VAR Volt-Ampere Reactive

WAN Wide Area Network

Symbols

$\boldsymbol{C}$	[F]	Capacitance
$\boldsymbol{C_f}$		Crest factor
$\boldsymbol{C_{bat}}$	[Ah]	Nominal battery capacity
$\boldsymbol{D_f}$		Demand factor
$\boldsymbol{E}$	[Wh]	Electrical energy
$\boldsymbol{FF(\cdot)}$		Fitness function
$\boldsymbol{L}$	[Hr]	Inductance
$\boldsymbol{L_C}$	[%]	Controllable load capacity
$\boldsymbol{L_f}$		Load factor
$\boldsymbol{P}$	[W]	Electrical power
$\boldsymbol{P(t)}$	[W]	Instantaneous aggregated consumption power
$\boldsymbol{PV_p}$	[%]	Photovoltaic electricity penetration
$\boldsymbol{P^c(t)}$	[W]	Instantaneous controllable consumption power
$\boldsymbol{P^{nc}(t)}$	[W]	Instantaneous non-controllable consumption power
$\boldsymbol{P_B(t)}$	[W]	Power storage in the battery
$\boldsymbol{P_G(t)}$	[W]	Power exchange with the grid
$\boldsymbol{P_L(t)}$	[W]	Power consumed by the loads
$\boldsymbol{P_{PV}(t)}$	[W]	PV power generated
$\boldsymbol{Q}$	[var]	Reactive power
$\boldsymbol{R}$	[Ω]	Resistance
$\boldsymbol{S}$	[VA]	Apparent power
$\boldsymbol{W_K}$		Time window of K samples
$\boldsymbol{\Delta t^u_{i,j}}$	[m]	Running range of a deferrable load defined by the user
$\boldsymbol{\Phi}$	[rad]	Phase difference of v and i
$\boldsymbol{\alpha_i}$		Random learning rate for the *ith* neural controller
$\boldsymbol{W}$		Matrix of synaptic weights connections
$\mathbf{cos(\Phi)}$		Power factor
$\boldsymbol{\hat{P}^{nc}(t)}$	[W]	All non-controllable consumption from local perspective
$\boldsymbol{\mu}$		Mean value of a signal in a period of time

$\boldsymbol{\mu_3}$		Skewness of a signal in a period of time
$\boldsymbol{\mu_4}$		Kurtosis of a signal in a period of time
$\boldsymbol{\mu_s}$		Mean value of $s(t)$ in a W_K
$\boldsymbol{\nu_i}$		Postsynaptic potential of the ith neuron
$\boldsymbol{\overline{C_f}}$		Average crest factor
$\boldsymbol{\overline{D_f}}$		Average demand factor
$\boldsymbol{\overline{L_f}}$		Average load factor
$\boldsymbol{\sigma^2}$		Variance of a signal in a period of time
$\boldsymbol{\sigma_s^2}$		Variance of $s(t)$ in a W_K
$\boldsymbol{\sigma_i(\cdot)}$		Activation neuron of the ith neuron
$\boldsymbol{\tau_i}$		Time constant of postsynaptic node
$\boldsymbol{\theta_i}$		Threshold of the ith neuron
$\boldsymbol{\xi}$		Self-consumption factor
$\boldsymbol{\xi_G}$		Normalized ξ by the generation
$\boldsymbol{\xi_L}$		Normalized ξ by the consumption
$\boldsymbol{c_v}$		Coefficient of variation
$\boldsymbol{f}$	[Hz]	Frequency
$\boldsymbol{f_{L,i}(t)}$		Local consumption pattern
$\boldsymbol{i}$	[A]	Electric current
$\boldsymbol{p_i^c(t)}$	[W]	Instantaneous controllable consumption of the ith user
$\boldsymbol{p_i^{nc}(t)}$	[W]	Instantaneous non-controllable consumption of ith user
$\boldsymbol{pdf(t_{i,j}^{act})}$		Probability density function of $t_{i,j}^{act}$
$\boldsymbol{s(t)}$		Aggregated consumption signal
$\boldsymbol{t_{i,j}^{act}}$	[m]	Starting time of a load defined by the user
$\boldsymbol{t_{i,j}^{beg}}$	[m]	Lower bound of $\Delta t_{i,j}^u$
$\boldsymbol{t_{i,j}^{end}}$	[m]	Upper bound of $\Delta t_{i,j}^u$
$\boldsymbol{t_c}$		Time of convergence
$\boldsymbol{v}$	[V]	Electric potential difference
$\boldsymbol{w_{ij}}$		Synaptic weight of the ith neuron for the jth input
$\boldsymbol{x(t)}$		Controllable demand signal
$\boldsymbol{x_j}$		The jth input of the ith neuron
$\boldsymbol{y_i}$		Output of the ith neuron
$\boldsymbol{z(t)}$		Non-controllable demand signal

1 Introduction

"Sapere aude" — Horacio

1.1 Motivation and Problem statement

Nowadays electricity is an essential element in daily life. The growth in electricity demand is driven by the increasing use of electrical and electronic devices (e.g. mobile phones, computers or appliances). In addition, the trend in electricity consumption is also positive[1] (see Figure 1.1). According to the International Energy Agency (IEA), from 2012 to 2040, the global electricity demand is projected to increase by 2.1 % per year in the current policy scenario (IEA, 2014d). This growth rate is the result of rising standards of living, economic expansion and the continuous electrification of society.

For some applications, such as electronic appliances, electricity is the only available option to enable them to perform their function. Furthermore, new electronic devices traditionally dependent on fossil fuels are emerging, such as the Electric Vehicle (EV) or new forms of electricity generation. Electricity offers a variety of services, often in a more practical, convenient and effective way than alternative forms of energy. In addition, electricity produces no waste or emissions at the point of use and it is available to consumers immediately on demand.

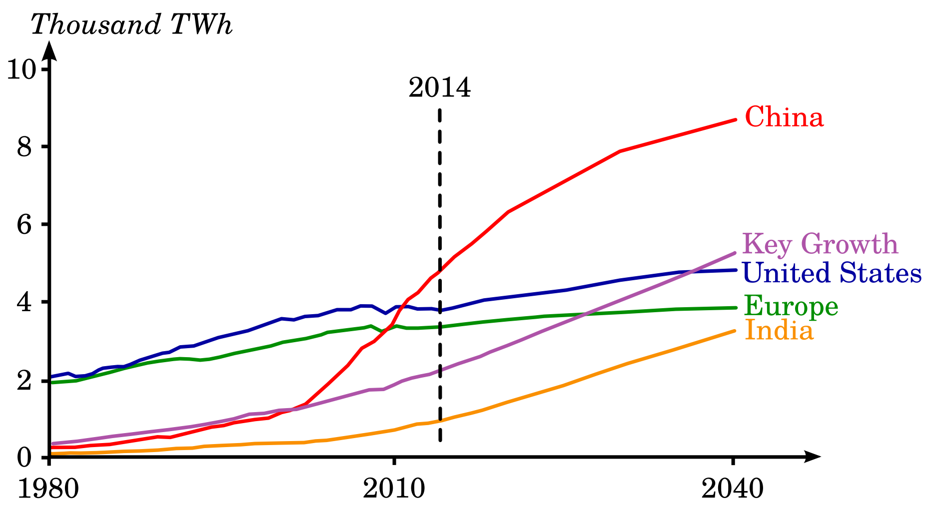


Figure 1.1: Trends of the demand growth in the world. Source:*Exxon Mobil.*

[1] http://corporate.exxonmobil.com/

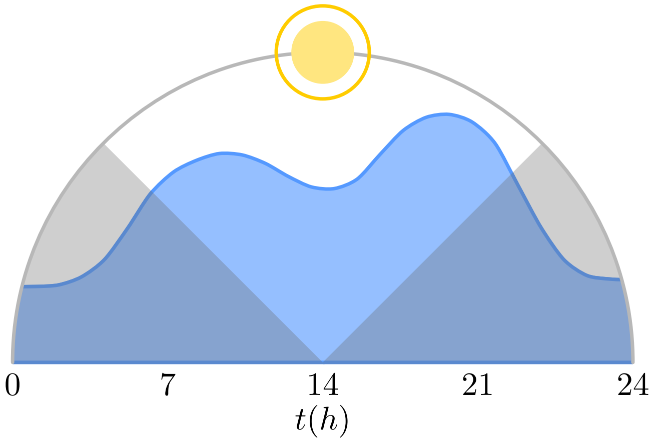


Figure 1.2: Daily aggregated consumption of an electrical grid. In shadow grey the night period is represented.

However, the grid must satisfy a single and problematic rule, the electrical consumption must be equal to the generation. This rule, in spite of being simple, is the source of the main problems of the grid. Grid operators must guarantee the security of supply to the users. If the generated power is bigger than the consumed power, the voltage and the frequency of the grid are greater than the operation point (e.g. 230 V and 50 Hz in Spain). On the other hand, if consumption is bigger than generation, the voltage and the frequency of the grid decrease their values. These variations can cause damages on both sides, generation and consumption. Therefore, generation and consumption must always be balanced in real time. This restriction requires a high synchronization between thousands or even millions of devices.

Aggregated consumption is defined as the sum of all loads inside the grid that consume power. Figure 1.2 shows an example of the typical waveform of a grid aggregated consumption. Generally, consumption is not constant throughout the day, leading to wide variations in power generation. Low consumption periods are called valleys and higher consumption ones are called peaks (see Figure 1.2). For example, in Spain according to Red Eléctrica de España (REE)[2], the valleys are nowadays in the vicinity of the 22-24 GW and peaks in the vicinity of the 36-38 GW. The installed generation capacity should be able to meet the peaks plus a safety margin. This means that during periods of low consumption, a small percentage of the installed generation is used (the average utilization of generation capacity is less than 55 %). However, an electrical grid is designed for the worst possible scenario and all its elements should have enough capacity to supply the maximum historical peak. Thus, the total capacity of the system is only supplied for a few hours in the year and these resources are underutilized most of the time. This oversizing of the grid in generation, transmission and distribution implies an inefficient structure whose investments and costs are high.

In addition, actual grids face significant regional disparities. This problem arises because of a centralized generation model, where electricity is generated far from the consumption points (see Figure 1.3, old generation grids). Indeed, this model of operation implies an electricity loss rate in transmission and distribution of approximately 9 % on average around the world, according to the IEA (IEA, 2014d). Furthermore, grid congestion happens in areas where consumption is higher than the available generation and it is produced because of the saturation of transmission and distribution lines, increasing the inefficiency of the grid. Another factor to be taken into account is that actual grids present low storage capacity, only a few forms are integrated in the system such as pumping hydro. The main advantage of incorporating storage systems to the grid is the possibility to defer the generation

[2] http://www.ree.es/es/

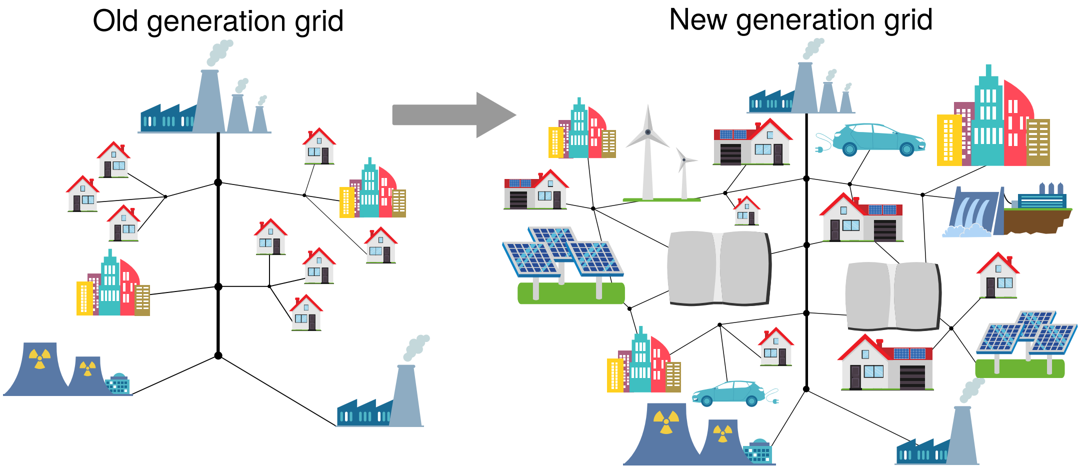


Figure 1.3: Old generation grids vs. new generation grids.

surplus to supply the demand when it is needed. The lack of storage systems causes certain risks in security and quality of electricity supply and increases the oversizing of the infrastructure. Finding a solution to the above problems is becoming a priority in energy policies worldwide (IEA, 2014d).

The new generation grids, known as Smart Grids (SGs) (see Figure 1.3, new generation grids), aim to solve these problems. This type of grids appears as a convergence between Information and Communications Technologies (ICTs) and engineering power systems (Farhangi, 2010). SGs started in the late 90s as an attempt to use electronic control and monitoring in the increasingly complex electric power systems (Vu et al., 1997). SGs have included since then, other concepts besides the reliability of the system, as for example advanced monitoring systems, better demand management, efficiency of the electricity transmission, self-healing, etc. Within SGs, the grid can achieve greater penetration of renewable technologies such as Photovoltaics (PV), and new manageable electrical consumption devices, as well as EV or automated appliances. Moreover, Distributed Generation (DG) would change the generation paradigm inside the SG thanks to the use of the ICTs. DG follows a different generation structure where small generators are spread over the grid and closer to the consumption. The interest in DG has been growing for 20 years in all participating collectives of the grid (Lopes et al., 2007). One of the main reasons is the reduction of the transport and distribution power losses. Another important benefit of DG is that it also reduces the purchase of external resources because the majority of technologies are based on renewable energies.

Another interesting area inside the SG, is the field of Demand Side Management (DSM), in which significant efforts have been made. DSM can be defined as actions that influence the way users consume electricity in order to achieve an objective, such as higher savings or energy efficiency. DSM achieves the following benefits: reduce the oversizing of the infrastructure, increase the profitability of investments in the grid, security enhancement, integration of new generation technologies and integration new local consumption technologies (Strbac, 2008). Furthermore, DSM mechanisms can be of different nature: changes in the regulatory environment, consumer awareness, efficiency of electrical equipment, billing systems, etc. (Torriti et al., 2010). The aggregated consumption presents a high variability in its waveform as shown in Figure 1.2, and it is one of the major problems of the current grid. The reason is that the complexity of the grid operation is increased as forecasts are needed to meet the demand. In general, DSM techniques aim at modifying the aggregated consumption to flatten its shape and increase the efficiency of the whole system. For these purposes, DSM techniques can be divided in four main techniques:

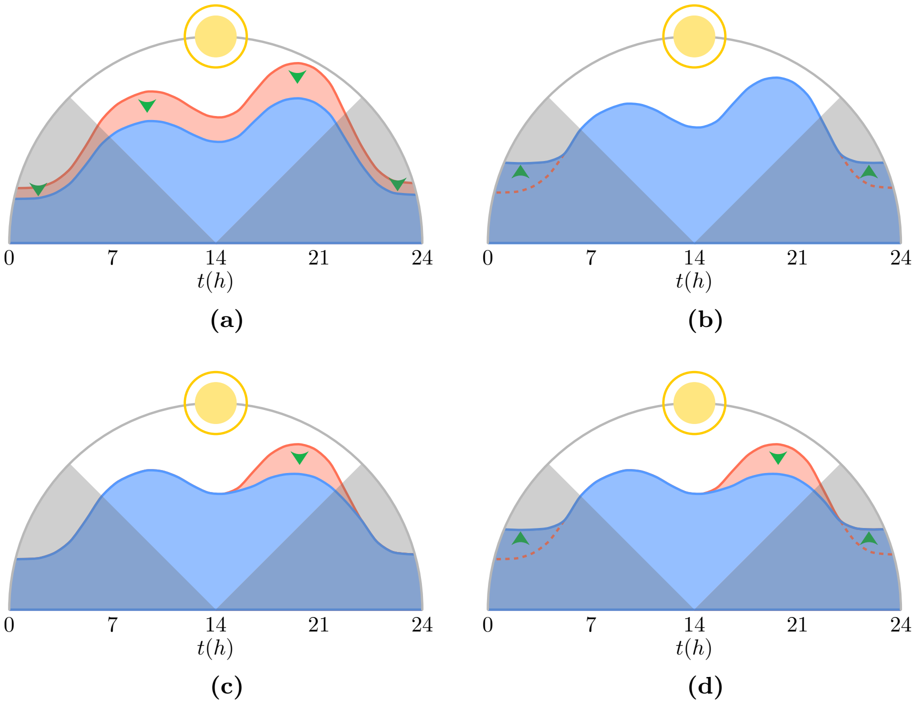


Figure 1.4: Effects of DSM techniques in the aggregated consumption: (a) consumption reduction, (b) valley consumption increase, (c) peak consumption decrease and (d) load shifting. In red and blue are the aggregated consumption before and after applying DSM techniques respectively and in shadow grey night periods are represented.

i) *Consumption reduction*, without modifying the shape of the aggregated consumption (see Figure 1.4(a)). The reduction is achieved by improving the efficiency of equipment and processes or social awareness.

ii) *Valley consumption increase*, more consumption during these periods (see Figure 1.4(b)). In this case, some new consumptions are added to the aggregated consumption, such as pumping stations, storage systems or electric vehicles, among others.

iii) *Peak consumption decrease*, reducing the consumption through interrupting services or managing the load during these periods (see Figure 1.4(c)). In these cases, a part of the existing load has to be removed from the peak so this measure is more restrictive than the previous one.

iv) *Load shifting*, displacing part of the electric loads from peak to valley periods (see Figure 1.4(d)). It is a combination of the ii) and iii) techniques. Specific regulations must be proposed to promote the use of this technique, such as time discrimination or response to market prices.

All DSM techniques look for an enhancement of the system efficiency. However, technique i) only reduces the grid size and not the variability of the whole system. And techniques like ii) seek consuming more in the valleys to fill them. Although they enhance the grid efficiency, more consumption implies more generation. If the valley consumption grows too much without reducing other parts of the aggregated consumption, it could suppose a loss of efficiency. On the other hand, technique iii) is certainly increasing the efficiency of the grid, but the behavior of the users is being restricted. Thus, for these reasons, technique iv) tries to compensate all the disadvantages of previous techniques. Without adding more consumption to the system, it is able to reduce the variability of the aggregated consumption, considering the behavior of users without interrupting their supply and reducing the consumption during certain periods of time. Therefore, this last technique is emerging as an important incentive to include in the development of SG since it helps to improve the system's operation and means a better integration of the different parts of the grid.

In this Thesis, a solution to the aforementioned problems from an Artificial Intelligence (AI) perspective is proposed. Specifically, it is proposed the use of Artificial Neural Networks (ANNs) to manage the flows of energy inside the future SG. The use of ANNs is motivated by the advantages of working with distributed, adaptive and nonlinear systems. Inside the field of ANN, Recurrent Neural Networks (RNNs) have been selected because of their dynamic behavior, which fits perfectly with the non linear dynamic behavior of the grid. In addition, a neural controller is developed to operate in each element of the grid, in order to increase its global efficiency by smoothing the aggregated consumption. Inside this grid environment, it is also contained DG, particularly PV because it is the only renewable generation technology liable to be widely used in consumption points. Thus, the effects of local supply inside the grid and the penetration of PV without destabilizing the grid behavior are studied. Finally, it is worthy to mention that the sole information available from the grid is the aggregated consumption. The neural controllers make decisions based on this sole information.

1.2 Thesis aim

The aim of this Thesis is the development of an adaptive algorithm to manage the consumption of a collective of individuals with the presence of Distributed Energy Resources (DER), which combines the DG with storage systems. The objective of this algorithm is to enhance the efficiency of a grid by reducing the variability of the aggregated consumption through its smoothing. The only information available for the algorithm is the aggregated consumption coming from the grid. With this information the algorithm must decide when it is the best time to consume aiming to accomplish the objective of smoothing the curve. Thus, the algorithm with the historical data of the grid has to predict and adapt the consumption of several individuals to fill the valley and decrease the peaks. As a whole, the collective of individuals must produce a flat aggregated consumption. Thus, the distributed self-organized algorithm senses the global behavior of a grid and only changes the local behavior of each individual inside the collective. Therefore, this Thesis seeks for a distributed DSM approach combined with an automatic control of demand that helps to integrate DER (DG and Electrical Energy Storage (EES)), which leads to an innovative concept called Active Demand Side Management (ADSM).

In addition, the grid is a highly non linear and dynamic system, due to the large number of elements and variables involved. For these reasons, ANNs have been chosen to develop the algorithm. ANNs have a high ability to learn from the environment and all information is processed in a distributed manner (Haykin, 2009). More precisely, RNNs, a variant of ANNs, present the necessary features to model such a complex system. The use of these algorithms allows taking advantage of the RNN properties such as non-linearity, adaptability, dynamics, distributivity, robustness, etc. These are enough advantages to develop an algorithm based on RNN.

From the energy efficiency point of view, the application of DSM techniques is able to enhance the grid performance. In this way, the oversizing of the grid and waste of the existing resources inside the grid are avoided. In addition, it is becoming more common to find DG inside grids because of the increase of solutions and their benefits. Hence, this Thesis seeks to take advantage of the energy resources available locally through PV DG. However, the local availability of electricity to supply the local demand could increase the variability of the aggregated consumption. This fact concerns to grid operators, who think that a high penetration of PV generation could be detrimental to the grid stability. Therefore, this Thesis tries to solve the PV integration problem, mitigating the stability grid issue and increasing the penetration of this source of electricity.

In the design of the ADSM algorithm, the following features have been taken into account:

- *Adaptivity*, it has to react quickly to changes in the aggregated consumption and adapt its output to counteract them.
- *System dynamics*, the grid is a complex system to model, so that the algorithm has to understand its nonlinear behavior over time by using the flows of energy, internal feedback loops and time delays.
- *Robustness*, it has to tolerate perturbations that might affect its functioning or the grid.
- *Distributed system*, the algorithm operates locally at the individual level, but the result of the whole should be flattened aggregated consumption without interactions or passing information among them.
- *Scalability*, it must be able to operate with any number of users inside the grid. This implies that it has to allow the incorporation of new users without interfering with its operation.
- *Data availability*, due to the privacy of the users inside the grid, the only information available is the aggregated consumption and the local consumption where it is operating. There is no information exchange among the individuals. This is a restriction due to the way that the data privacy is treated inside the actual grids.

1.3 Thesis structure

This Thesis is divided in three main parts. In Part I, the basic concepts of this Thesis are introduced to provide the reader with the necessary context to understand the problem stated and the proposed solution. Chapter 2 gives an introduction to the energy concepts related with the grid. The ANN concepts and description of the elements used to develop the ADSM algorithm are described in Chapter 3.

In Part II, the proposed ADSM algorithm is described. The development of the algorithm has been divided in three different stages based on simplifications of the grid environment. Chapter 4 proposed a first approximation of the problem based on a reduced environment of two users, one controllable and one non-controllable. The neural controller parameters are tuned in order to solve this simplification of the problem and an analysis of its performance is developed. Then, in Chapter 5, the problem is taken to the next step and an algorithm is developed to coordinate the response of the neural controllers in a collective environment where the behavior of controllable and non-controllable users is analyzed. Chapter 6 summarizes the results of using the proposed algorithm in a simulated grid environment.

The conclusions and future works are collected in Part III with Chapter 7. Section 7.1 reviews the work done in the Thesis and discusses its main aspects. Section 7.2 suggests a list of proposals and improvements of the proposed algorithms. Finally, a review of the author related contributions is summarized in Section 7.3.

PART I

Background

2 Grid Framework

"All power corrupts, but we need the electricity" — Anonymous

Electricity has become an essential part of life. Like air, for most users it is a transparent fact in their lives, of which they have no notice. It is only when power disappears during a failure, when people realize how important electricity is in daily life. Electricity is used to power computers, mobile phones, cooling, cooking, washing clothes, lights, entertainment, transportation, etc. (Brain and Roos, 2000). But, how is it possible that electricity is there when switching on a light? How is electricity able to reach every place instantaneously? The answer to both questions is the electrical grid or only the "grid". A grid is a network of electrical power systems which operates in real-time (Blume, 2007). This means that power is generated, transported and supplied in the same instant that it is needed. The main rule of the grid is that generation must match consumption. Electric power systems do not inherently store energy such as water or gas systems. The reason is in the nature of electricity which is produced by the movement of electrons. Instead, generators produce the energy as the demand calls for it.

Thus, the grid is a vast physical and human network connecting thousands of electricity generators to millions of consumers — a linked system of public and private enterprises operating. However, it had only minor changes in its structure for the past century. So, the grid will face different challenges over the next decades, while new technologies arise as valuable opportunities to meet these challenges (MIT, 2011). Some of them are already here such as integration of renewable energy, Distributed Generation (DG) or smart metering. If grid operators fail to realize about these challenges, it could result in degraded reliability, significantly increased costs, and a failure to achieve several public policy goals (MIT, 2011). To overcome these drawbacks, it is necessary to take the grid to the next level. The Smart Grid (SG) is conceived to address the global challenges of energy security, climate change and economic growth. SGs have some features that enable several low-carbon energy technologies, including Electric Vehicles (EVs), variable renewable energy sources and Demand Side Management (DSM) (IEA, 2011).

In this Chapter, the current status of the electric system is introduced (see Section 2.1), the main parts in which is divided, such as generation (Section 2.1.1), transmission (Section 2.1.2), substations (Section 2.1.3), distribution (Section 2.1.4) and consumption (Section 2.1.5). Thus, the present form of the grid is introduced together with the identified issues that operators have to tackle within the evolution towards the next grid generation. These problems will be reported in Section 2.2. After introducing all these concepts, explanations about what the SG is and the changes involved in its deployment, both structural (physical) and internal (logical), are presented. Thus, in Section 2.3, the concept of the SG is defined and the key aspects on which it is founded are introduced: Information and Communications Technology (ICT) (see Section 2.3.1), DG (Section 2.3.2), Electrical Energy Storage (EES) (Section 2.3.3) and DSM (Section 2.3.4). In this Thesis, the use of DSM techniques are emphasized in order to enhance the grid efficiency integrating renewable generation and EES. All of them are necessary for the proper growth of a development environment and proper operation of these new generation power grids.

2.1 The electric power systems

The electric power system is one of the most complex man-made systems. It consists of a set of elements that operate coordinately in a territory to satisfy the demand. The first electric power system was introduced in 1882 by Thomas Edison. It consisted of 59 customers connected in Direct Current (DC) at a price of about 5 $/kWh (MIT, 2011). The development of the grid has been made over the last century and a half. Initially, the electric power systems supplied different, small and isolated consumers. And, the grid was born of the interconnection of these systems as a result of demand growth (Schewe, 2007). This interconnection was favored by two factors: i) the transition from DC to Alternating Current (AC) power and ii) the use of transformers to elevate voltage, making transportation possible with fewer losses (MIT, 2011). AC power systems replaced DC ones because they present more advantages: easily elevation of the voltage level and generators and motors are much simpler (Boyle, 2007). During the next decades, new elements were introduced in the grids that improved the quality as consumption continued growing. With these achievements together with the growth of the grid, utilities could take advantage of economies of scale and the price of electricity decreased its value.

Before the 1990s, individual companies controlled all the parts inside the grid from generation to delivery for a geographic area. In the 1990s, the regularization of electricity supplies changed, using markets rather than regulators to set prices, where electricity is treated as a commodity (Fox-Penner, 2010). Deregularization separated the generation into an industry apart and the electricity was produced far from the consumers. This fact introduced more complexity and interconnectivity within the grid. The more complex the grid becomes, the more instabilities affect the grid. A failure in one part of the grid can be propagated to the remaining parts. Along with the aging of the infrastructure, power losses increased and maintaining them is increasingly expensive.

Despite the inclusion of new developments within the grid structure, it has not changed its main internal organization in about a century and a half. Electrical grids around the world have a vertical structure, which interconnect generators, substations, transmission networks, distribution lines and consumption. Figure 2.1 shows the basic building blocks of the grid, in which the different elements mentioned before are shown (Kirtley, 2010).

- *Generation* is the part of the grid in charge of producing electricity. There are different types of generators that convert primary energy sources, such as fossil fuels and renewable resources, into electricity, e.g.: nuclear power plants, combined cycle plants, wind plants, Photovoltaics (PV) solar plants and hydro power plants among others. The size of the generators varies from few kilowatts of small diesel generators to thousands of megawatts of nuclear power plants.

- *Substations* are in charge of conditioning the electricity power between elements of the grid. These centers are responsible for raising or lowering the voltage (high/medium voltage and medium/low voltage) to couple different grid sections. Hence, they contain different elements such as transformers, electrical buses, capacitor banks, etc. In addition, substations can operate in the grid through protective relaying, breaker controls, metering, etc.

- *Transmission network* transports large amounts of electricity from generators to substations located close to the consumers. Almost every transmission line is high-voltage, three phases and AC. Electricity is transmitted at high voltages to reduce the energy losses over long distances. The electricity power parameters (voltage, frequency and number of phases) are regulated by the organ in charge of managing the grid, usually at country level.

- *Distribution network* transfers the electricity from the substations to its destination. The distribution lines operate at medium/low voltage depending on the consumers requirements. For instance, the electricity could be distributed

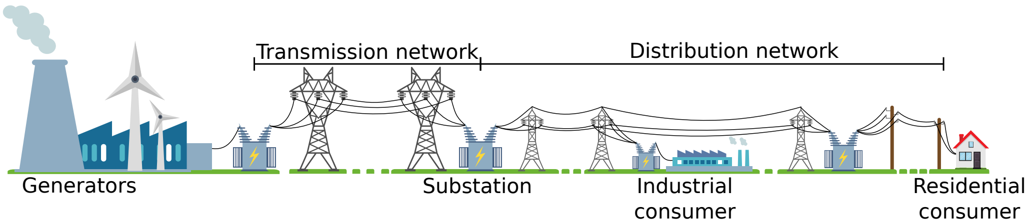


Figure 2.1: Block diagram of an actual electrical grid.

in one or more phases (typically up to three phases). In addition, the electricity lines that reach homes provide generally a single phase of 230 V in Europe.

- *Consumption* is considered as the electrical energy used by all the elements inside the power system. Thus, any device that transforms electric power into work is considered as consumption of the system. Consumption is made up of different heterogeneous elements that are grouped according to the final behavior of the grid users or consumers, e.g.: industrial, commercial or residential. Losses due to different elements within the grid are also considered as consumption.

These basic elements are part of different power systems. On the other hand, the grid electrical signal mainly consists of three physical characteristics: i) current, ii) voltage and iii) frequency (Blume, 2007). The electric current (i) is the amount of electric charge that flows inside a conductor per unit of time. Its unit in the International System of Units (SI) is the ampere (A). The electric potential difference or voltage (v) is the work applied to move electric charges between two points. In other words, it is the force that electrons need to move. Its unit of measurement in SI is the volt (V). Frequency (f) is the number of times a signal is repeated in certain time. Its unit of measurement is Hertz or hertz (Hz). For example, Europe has a value of 50 Hz, while in North America it is used 60 Hz. In the rest of the world, the frequency takes one of these two values depending on the country. All of these parameters are monitored by the grid operators to guarantee the stability of the electric system. The grid must always provide the required electricity that meets the varying demand. The reason is that electricity is consumed at the same time that it is generated. An imbalance between supply and demand will damage the stability and quality (voltage and frequency) of the power supply (IEC, 2011). But how are these parameters affected by that mismatch? If demand is greater than generation, frequency and voltage values fall while if generation is greater than demand, frequency and voltage rise. The grid must remain safe and be capable of withstanding a large variety of disturbances to guarantee a reliable service. For these reasons, grid operators design different contingency plans and constant surveillance from control centers (Kundur et al., 1994). One of the most widespread measures to achieve the right balance between generation and consumption is making electricity demand forecasts. With these forecasts, power plants prepare their production programs for each of the day hours to meet that demand (Boyle, 2007).

In spite of being very dependable from the geographical area where the grids are built, they present some features in common:

- *Ageing system*, since the deployment of the electric grids, there have been no deep changes in their infrastructure. It is not enough with this feature to meet the needs of users as well as complicate the entry of new elements within the grid. Therefore, it is necessary that the grid evolves at the same time that users increase their demands and technological evolution brings to the market new generation technologies.

- *Large scale*. The grid must be able to supply different number of users, ranging from just a few to millions, depending on the size of the area to cover. This fact

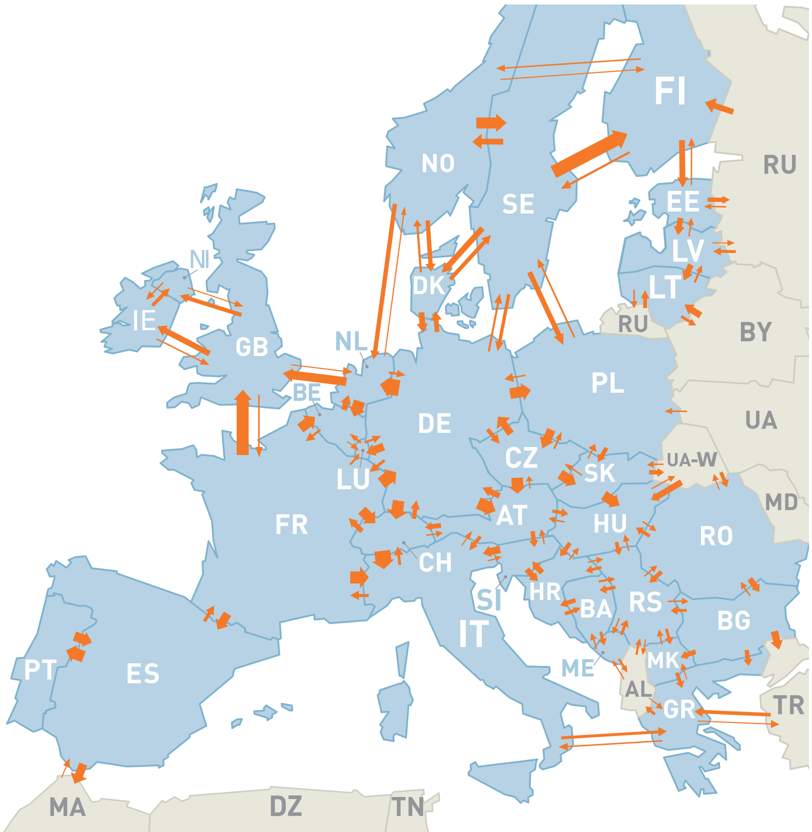


Figure 2.2: Map of electric exchanges for the different European countries. The thickness of the different arrows represents the amount of power exchanged between the different European countries.

raises the serious problem of avoiding potential performance degradation as the grid size increases.

- *Geographical distribution.* The grid is deployed along the length of the different countries where it is installed, covering most of the population. There may be one or more operators responsible for supplying the demand. These operators install the elements to guarantee the access to electricity, power lines, control centers, substations, etc.

- *Interconnected power systems.* The interconnection among electric systems allows guaranteeing the supply in a given territory when a particular system cannot generate enough power to meet demand. This happens when extraordinary and unexpected consumptions occur (e.g. a cold snap), or when one or several generation centers are no longer operating temporarily and enough electricity is not injected into the system. For this reason, the more electrical systems are interconnected and the higher the capacity of energy exchanged is, the greater the safety and quality of service they provide. For example, the continental European electricity system is connected to the Nordic countries and the British Isles by North and Eastern countries (see Figure 2.2).

- *Heterogeneity.* A grid hosts elements whose nature is very different. Within the grid there are different generators types, different power lines, different types of consumers, etc. There are also differences between the physical features of the electricity supplied, different frequencies (50 Hz or 60 Hz) or different voltage

(120 V or 230 V). The values of these parameters depend on the region in which the grid is located.

- *One-way communication.* Consumers are mostly uninformed and do not have active part in the system, they only demand electricity. The communication goes therefore from generators to consumers through control centres of the grid operators. However, if users get involved in the process, the efficiency of the system would be enhanced. The reason is that their behaviour affects directly to the grid status and the modifications of this behavior increases the grid operation effectiveness.

- *Multiple agents.* Each grid may establish different security and administrative policies under which a safe electricity market can be developed and would be profitable for all participants. As a result, the already challenging grid security problem is complicated even more with the inclusion of new communication technologies.

- *Resource coordination.* Resources in a grid must be coordinated in order to provide aggregated capabilities. Thus, the generation could meet the demand at any time without any problem.

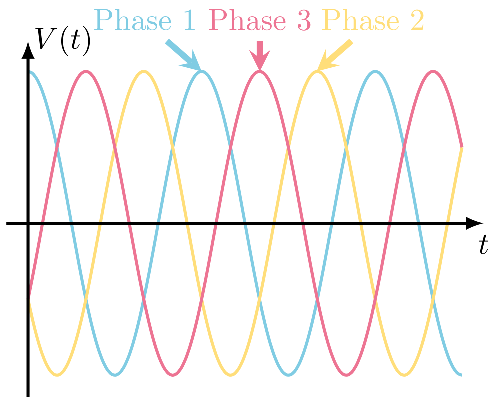
- *Access* to the grid must be: i) transparent, the grid should be seen as a single system, ii) reliable, the grid must guarantee the supply to the users at any moment under established quality of services requirements, iii) consistent, the grid must bring together all its constituent elements to supply the demand, and iv) universal, the grid must grant access throughout the whole operation territory and adapt itself to a dynamic environment that changes continuously.

Nowadays, the integration of new generation technologies into the grid is becoming complex due to its age. New technologies powered by renewable resources such as wind and solar energy are increasingly becoming part of the generation mixture with considerable difficulties because of the variability of these energy resources (Boyle, 2007) and the centralized generation paradigm on which electric grids are based. In order to include these resources, it is necessary to use sophisticated management algorithms that can meet the demand in real-time. Thus, the algorithms are constantly changing the supplies of energy from different sources to tackle the variability of renewable energies. Collecting data from the different elements of the grid could help to perform the complex calculations for grid stabilization. On the other hand, a bad management of the collected data supposes a curse to the system. It is in this scenario in which the SG arises. The concept of the SG will be explained later in Section 2.3. But first, each grid component is explained in more detail.

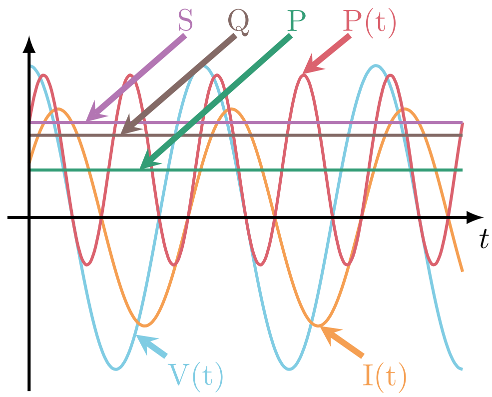
2.1.1 Electricity generation

Generators are the part of the grid in which the electricity is generated to supply the demand. They convert primary energy sources into electric energy. This primary energy comes from different sources, such as fossil fuels, uranium, wastes, water, wind, the Sun, etc. Then, to transform these primary energies into electricity, the process differs depending on the generating unit, whose design is defined by the raw material used (Blume, 2007).

The power generated must have the characteristics imposed on the grid interconnection point by the operators. In general, an AC electric generator produces alternating power at its terminals (Kirtley, 2010). The generator possesses three terminals, which provide AC voltage and current in each of them. Those electric signals are 120 degrees out of phase with respect to each other, as shown in Figure 2.3(a). This set of signals is known as three-phase AC voltage. Three-phase AC is used due to some advantages, such as requiring less conducting material in the transmission lines or allowing a constant power flow from the generators (Schewe, 2007). Focusing on a single phase, the power generated instantaneously ($P(\mathrm{t})$) can



(a)



(b)

Figure 2.3: AC physical signal: (a) three phase voltage and (b) power signals.

be defined as the flow of electrical energy (E) in a circuit or the work done per unit of time, as shown in Equation 2.1.

$$P(t) = v(t) \cdot i(t) \tag{2.1}$$

where, v(t) and i(t) are the instantaneous values of the electric voltage and current per phase respectively. In AC systems, both v(t) and i(t) vary periodically so the resultant P(t) is also an oscillating signal (see Figure 2.3(b)). However, the rapid variation does not provide a reliable measure of P(t), so that an average value is estimated by evaluating several cycles. In fact, the parameters of special interest in power systems are active power (P), reactive power (Q) and apparent power (S) (see Figure 2.3(b)):

- P, active power or real power, is the power that performs useful work. It is measured in watts (W) and the mathematical expression is described in Equation 2.2a. In this Equation, the v_{rms} and i_{rms} are the Root Mean Square (rms) values of instantaneous voltage and current respectively and Φ is the phase difference of v and i.

- Q or reactive power is the amount of power that does no useful work and it causes losses in the system. It is measured in volt-amperes reactive (var) and it is calculated as in Equation 2.2b.

- S is the combination of P and Q, or simply the product of v_{rms} and i_{rms} (see Equation 2.2c). It is measured in volt-amperes (VA). S is always greater than or equal to P and Q (Kirtley, 2010).

$$P = v_{rms} \cdot i_{rms} \cdot \cos(\Phi) \tag{2.2a}$$

$$Q = v_{rms} \cdot i_{rms} \cdot \sin(\Phi) \tag{2.2b}$$

$$S = \sqrt{P^2 + Q^2} = v_{rms} \cdot i_{rms} \tag{2.2c}$$

The $\cos(\Phi)$ or power factor is of special interest and describes the ratio of P to S. The Φ angle occurs due to the reactances of the different loads in the system, e.g. transmission lines. Its value has a direct influence on power. If v(t) and i(t) are in phase, they are reaching the same stages at the same time. Despite being varying, P(t) is always positive or transmitted in the same direction and all the power is performing work, P. Nevertheless, in case v(t) and i(t) are shifted (out of phase), P(t) takes positive and negative values. In this case, the power is not only flowing

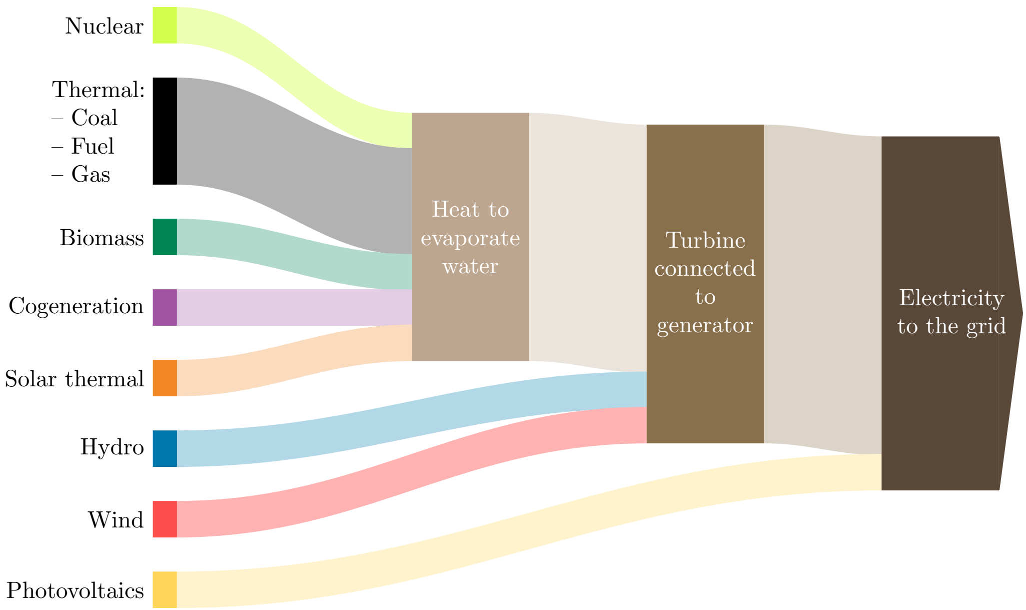


Figure 2.4: Forms of electricity generation.

in one direction (P), there is also a back and forth movement (Q). The sign of P(t) indicates its direction, but for P and Q, the sign indicates the phase shift of v(t) and i(t). Because Q produces no useful work, power systems try to compensate Φ by using specific loads to correct this behavior. Grid operators want a $\cos(\Phi) = 1$, because it implies that all the generated power is performing useful work (von Meier, 2006).

In order to generate electricity, the most widespread method consists of transforming mechanical energy in electricity through the movement of a turbine. That is the reason why electricity is normally called a secondary energy source. A turbine is a simple device with few parts that uses flowing fluids (liquids or gases), forcing them to pass through blades mounted on a shaft, which causes the shaft to turn. Then, the mechanical energy produced from the rotation of the shaft is collected by an AC electric generator which converts the motion to electricity by means of magnetic fields (von Meier, 2006). There are three main sources to supply a generator with mechanical energy: i) high pressure steam, ii) falling liquid water and iii) wind. Figure 2.4 shows the principal forms of generating electricity. Except for PV generators the rest of them include a turbine and a generator. So, different power plants can also be classified depending on the turbine used to transform the primary energy used (Blume, 2007):

- *Steam turbine*, it uses high-pressure and high-temperature steam to drive the mechanical shaft of the AC electric generator. The steam is created in a boiler, furnace, or heat exchanger and moved to the turbine. Depending on the fuel, there exist different power plants. Examples of power plants that uses steam turbine are nuclear, thermal, geothermal, solar-thermal and biomass, among others.

- *Hydro turbine*, it takes advantage of the water to generate electricity. Hydroelectric power plants convert the kinetic energy of falling water under the influence of gravity.

- *Combustion turbine* power plants burn fuel in a jet engine and use the exhaust gasses to spin a turbine generator. In general, it uses a mixture of a fuel (e.g., diesel fuel, jet fuel, or natural gas) and air. The power plants that use this type of turbine are known as combined-cycle power plant.

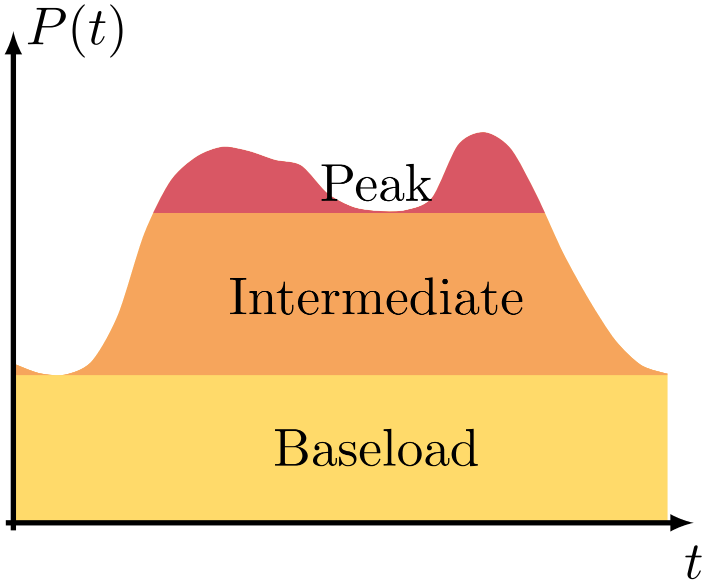


Figure 2.5: Operational classification of generation units.

- *Wind turbine* uses the kinetic energy of the wind to move directly the mechanical shaft of the AC electric generator.

Solar PV generation is the only generation technology that does not need the combination of turbine plus AC generator to produce electricity. PV consists of generating electricity from sunlight employing PV solar cells. These are large area junction diodes that produce current when sunlight shines on them and splits electron/hole pairs (Kirtley, 2010). The PV cells are grouped in series or parallel to form modules, elevating v and i respectively. These modules are grouped in systems that are connected to the grid. PV modules produce DC energy at their terminal outputs. It is necessary to use an intermediate element before injecting the generated electricity into the grid. This device is called inverter and transforms DC into AC electricity. Solar PV plants are environmentally friendly as they do not produce any harmful waste when generating electricity. The most efficient commercial modules have an efficiency around 20 % and the turn-key PV system costs are around 1.5–4.5 $^{\$}/_{W_p}$ (IEA, 2014b). PV growth has shown a positive trend, influenced by cost reductions and electricity prices increase. Another concern about PV energy is the efficiency, which is directly related to the space it occupies. Some critics accuse PV generators of having low efficiency and requiring large lands to obtain the same amount of electricity generated by other generation technologies. However, PV efficiency is also increasing and will reach higher rates in the near future (IEA, 2014b).

In addition, grid operators established another classification from the operational perspective of the power system. This classification divides the generating units in three categories (see Figure 2.5): i) baseload, ii) intermediate and iii) peaking units (Blume, 2007; Kirtley, 2010). Grid operators ensure that division depending on the unit costs. Baseload units are used to meet the constant power demanded in the system. They run continuously except when they need to be repaired or in maintenance, so that the generating unit is shut down. Thus, these generation units present two main requisites: reliability and reasonable costs. In general, nuclear and thermal coal plants are used as basaeload units due to their low fuel costs. However, those units present two main drawbacks: they are expensive to build and their output power can only be changed slowly for long periods of time (slow ramp rates). Another generation unit that can be considered baseload is the hydroelectric plant whose channel can be considered uninterrupted.

The intermediate units are also called cycling units. The output power of these units is regulated depending on the time of the day and it is extended for long periods of time. One feature that distinguishes this type of generation from baseload generation is the ability to vary their output power faster. Combined-cycle gas turbine plants and thermal generating units generally are used as intermediate units. Finally, peaking units only generate when the demand is close to the maximum or peak of the power system. They run only for a few hours per day, so they have to be able to start and stop quickly. Combined-cycle gas turbine and hydroelectric plants with reservoirs are generally used as peaking units. Combined-cycle gas turbines are the least expensive to build but have high operating costs.

In general, large generating units are located far from where their power is consumed, and the generated power has to be transported to load centers (Schewe, 2007). In order to reduce power losses during transmission, the generation voltage is raised from tens of kV to a few hundred using a transformer. Grid operators are constantly watching that generating units are synchronized (Kundur et al., 1994). Apart from large generating systems inside the grid, there exist also some small scale distributed generators, including combined heat and power units. Inside this small generating units, small hydroelectric plants, solar PV systems or small wind power generators are found. They operate at lower voltages and are connected at the distribution system level (Fox-Penner, 2010). Once explained the different forms of generation, it is necessary to transport this generated electricity to meet the user demand.

2.1.2 The transmission system

The transmission system is the part of the grid in charge of transporting electricity over long distances, from the place where it is generated to the distribution network. Recently, the generation is getting closer to the places where it is demanded. However, it is still far from the consumers. Furthermore, the transmission network ensures that the generated power comes with the least possible losses. This network is composed of different elements i) power lines, ii) high towers and iii) substations. The power lines are attached to the towers, which cover great lengths to fulfill their mission. However, insulated cables are normally used and buried underground when the power lines have to go through a population. Substations are in charge of conditioning the power transmitted and routing the power to its destination (see Section 2.1.3).

Different factors influence the transmission lines characteristics, such as current, material type, section, size, etc. These features will influence the amount of power that the line can transport inside it. The power line conductor presents a resistance (R) to the flow of current due to the properties of the material. This R makes the conductor to heat up, losing part of the transmitted power that circulates inside it. The conductor R is not constant with the length and it also varies with the diameter size of the conductor, being smaller the greater the diameter is (Blume, 2007). There are other effects present in the line due to the current. The flow of current through the line length produces magnetic fields which cause the appearance of an inductance (L) in series with the terminals of the line. Furthermore, a capacitance (C) appears in parallel to the transmission line terminals due to the appearance of electric fields for carrying voltage. Both, L and C depend on the power line length, the higher the length is, the higher the value of the parameters is (Kirtley, 2010). These three parameters, among others effects, are responsible for the losses in transmission lines. So it is very important to design properly the power lines to ensure the minimum losses in the power transmission. Some aspects to take into account in their design are: i) material, ii) conductor configuration, iii) section and iv) length (Molburg et al., 2007; Blume, 2007; Kirtley, 2010).

The transmission network has to transport power over long distances, so that High Voltage (HV) is used to reduce the losses. This transmission form reduces the conductor cross-sectional area and requires less right-of-ways for a given power (Blume, 2007). Transmission losses are mainly due to heat radiation by the conductors when the current flows inside. Thus, remembering Equation 2.1, the same amount of power could be transported if v is increased while i is decreased. In consequence, the

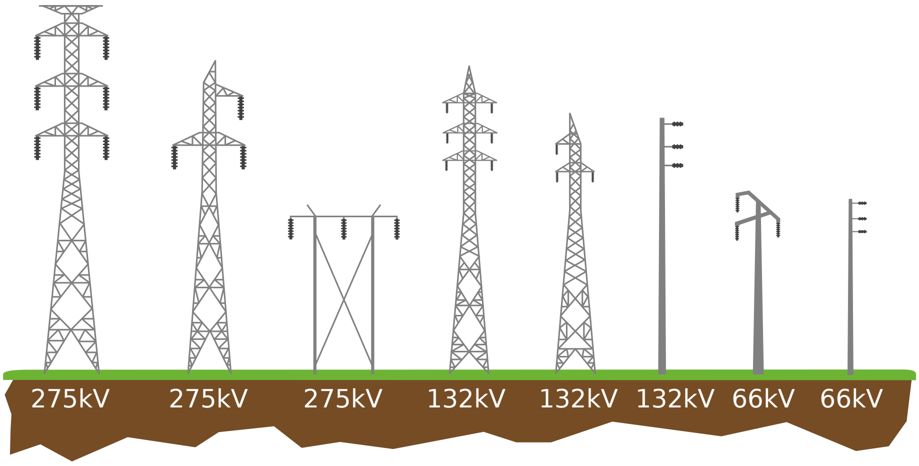


Figure 2.6: Different power transmission towers.

Region	LV (kV)	MV (kV)	HV (kV)	EHV (kV)	UHV (kV)
Europe	< 1	$1 - 132$	$132 - 380$	$380 - 700$	> 1000
North America		$1 - 72.5$	$132 - 475$	$500 - 800$	

Table 2.1: Voltage category for North American and European regions.

losses decrease drastically because they depend on the i^2 and also conductors need less section. Depending on the amount of energy transported and the distance traveled, each part of the grid carries electricity at a voltage or another. There exist different voltage classes depending on its value, i) Low Voltage (LV), ii) Medium Voltage (MV), iii) High Voltage (HV), iv) Extra High Voltage (EHV) and v) Ultra High Voltage (UHV) (Blume, 2007). Each voltage category has different values depending on the region. As an example, different voltage categories are gathered in Table 2.1 for North American and European regions. Moreover, the transmission voltages also vary within regions or countries. For example, the typical transmission lines used in the U.S. are 69 kV, 115 kV, 138 kV, 161 kV, 230 kV, 345 kV, 500 kV and 765 kV (von Meier, 2006). While for Spain, which is a smaller country in comparison, the voltages used are 60 kV, 110 kV, 132 kV, 150 kV, 220 kV and 400 kV[1]. The majority of the transmission lines are three phase HVAC, however for very long distances HVDC is being used (Schewe, 2007). Transmission in DC has some benefits, such as no synchronization issues and no reactive impedance because $f = 0$. It requires only two conductors instead of three. On the other hand, sophisticated inverter stations are required to transform AC to DC and vice versa in the interconnection point of both systems.

In order to cover the distances between the generation power plants and the distribution network, the power lines can be placed above or under the ground. In general, the widespread configuration used is above the ground attached to metal towers. Figure 2.6 shows different metal towers whose design varies depending on line voltage, conductor size and weight, tradition, etc. (von Meier, 2006). The power lines go underground when it is the only option left and aboveground transmission is not able to be installed (e.g. cities or airports).

The general topology of the transmission network is a mesh network, which consists of multiple paths to connect two points inside it. It is opposed to the radial network model in which points of the network are connected to a central node. One

[1] *Source:* REE

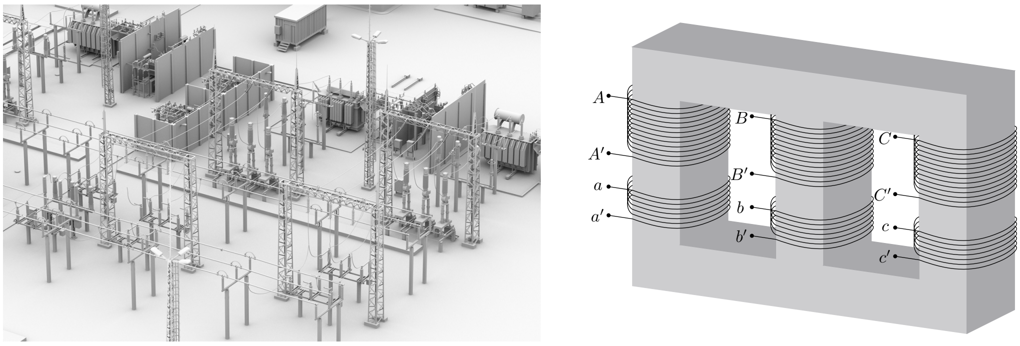
advantage of using a mesh topology consists of using its redundancy to guarantee security of supply. Thus, in case that a power line breaks down or a generation unit goes off line, the loads get the electricity they need. However, the path that the electric power follows cannot be easily known. In general, the only fact known is that electricity will flow from generators to loads by the lowest impedance path. Grid operators try to control the power flow for each line by means of forecasts, which requires a lot of computation and precise knowledge of the network. On the other hand, the presence of multiple paths inside the network leads to undesirable flows between them. Those paths are known as loop flows and they are produced because the current cannot be directed to any particular branch of the network (von Meier, 2006). Moreover, the power that the lines can transport are limited by three main factors among others, thermal, voltage stability and/or transient stability.

Finally, the continuing geographical expansion and interconnection of the transmission network is being motivated by technical, social and economic reasons. The interconnection brings the opportunity of economies of exchange, in which the sales of electricity grow with consumers inside the network. Another reason to interconnect them is the exploitation of economies of scale, in which the cost reductions of the use of electricity are higher as more users join to the grid. The cost is being shared by all the consumers, being able to expand and bring together more consumers. In addition, the interconnection favors the load factor, which represents the ratio of actual energy consumption over a period with the maximum power that can be instantaneously demanded. From the grid perspective, the ideal consumer would be demanding a constant amount of power all the time, although in reality it is not even close. However, grid operators search with the power system interconnection, grouping different types of consumers and larger amounts of power in order to smooth the consumption profile and approach to the ideal one. The last advantage of larger transmission systems is service reliability. With a greater degree of interconnection, if a generator cannot meet the demand, electricity can be supplied by others. There are also disadvantages of larger transmission systems size, such as energy losses or stability due to the interdependency between areas. However, the benefits of interconnection outweigh the drawbacks (von Meier, 2006).

2.1.3 Substations

Substations are the part of the grid that serve as interconnection point between different levels or sections of the power system. They have the ability to switch or reconfigure the ways in which power flows through the lines that cross them. Substations are spread over the transmission and distribution networks to connect power lines of the same or different voltages (von Meier, 2006). Substations are necessary for conditioning the power from the HV of the generators to the LV of the consumers. Thus, they exist at various scales to decrease the voltage in several steps before reaching customers. On the largest scale, there are transmission substations in charge of interconnecting different HV transmission power lines. Then, large substations at the intermediate scale interconnect HV transmission power lines to MV distribution power lines. Finally, at small scale, small substations are in charge of conditioning the power to cover the consumers necessities, and they serve limited localized areas. Depending on the size of the substations, the number of circuits that they can connect varies, from just a few to dozens of them. Large substations also posses a control room in which the proper operation of the systems is coordinated, whereas small ones are generally unstaffed (von Meier, 2006). In addition, substations provide protection for lines and equipment of the power system with different devices, that can be operated remotely or locally from the inside of the control rooms.

All the necessary equipment for regular system operation is found inside the substation (see Figure 2.7(a)) (MIT, 2011). Among this equipment, there are transformers, switchgear, measurement instrumentation and communication equipment. Transformers are used to change the voltage level at their inputs. The transformer is composed by a metal core and different coils of wire around it (see Figure 2.7(b)). The current flowing in the coil on one side of the transformer induces a voltage in the coil on the other side and both coils are coupled by the magnetic field. The



(a) (b)

Figure 2.7: Substation elements: (a) example of a substation and (b) scheme of a three phase transformer.

voltage between one side and the other depends proportionally of the turns' ratio of the coils, while the current is inversely proportional to that turn ratio. Other important elements are the switchgear which includes circuit breakers and switches. They disconnect parts of the network for system protection or maintenance.

Substations must also maintain the power flowing through them within a quality range. As power lines are longer, the drop in voltage increases, but the current also increases since the same amount of power is transmitted. This implies that also the losses increase which it is not desirable. Thus, substations must maintain the voltage within a specific range to ensure the quality of supply. The process of bringing voltage back within acceptable range is known as voltage support or Volt-Ampere Reactive (VAR) support. With voltage support, substations control power flow, improve transient stability on power grids, and reduce system losses. Compensating devices are used to provide voltage support: i) capacitor banks and/or ii) static VAR compensators.

Finally, substations also include measurement, communication and control equipment. Measurements are collected to monitor the status of the system. Then, those measures are sent to control centers where the data are processed to evaluate whether everything is working properly. If a fault occurs, the operators actuate over the different devices inside the substation to fix that fault. Thus, measurement instrumentation collects voltage, current, and power data for monitoring, control, and metering purposes. And communication equipment transmits these data, allowing switchgear to be controlled remotely. Substations do not only serve as interconnection points, but also as monitoring, protection and security points of the grid.

2.1.4 The distribution network

The distribution network is in charge of carrying the power from the transmission network to the consumers. The electricity is carried in wires that can be attached to poles or, in large populations, underground. Transmission networks are separated from distribution networks in terms of voltage level. The limit between them depends on the utilities which decide this value. The distribution network operates at lower voltages than the transmission network, so that it requires less clearance. Normally, the voltage ranges in which the distribution network works are in the MV and LV. Typically, this network is composed of lines up to 35 kV, but some lines reach higher values, around 70 kV (von Meier, 2006). Thus, smaller conductors are part of the distribution network because the distances to supply electricity are smaller.

As mentioned before, the transmission network is interconnected with the distribution network at distribution substations. Distribution lines are called *feeders* because they feed with electricity to the users. When distribution lines leave the substation, they carry three phase voltage. They use smaller conductors because the

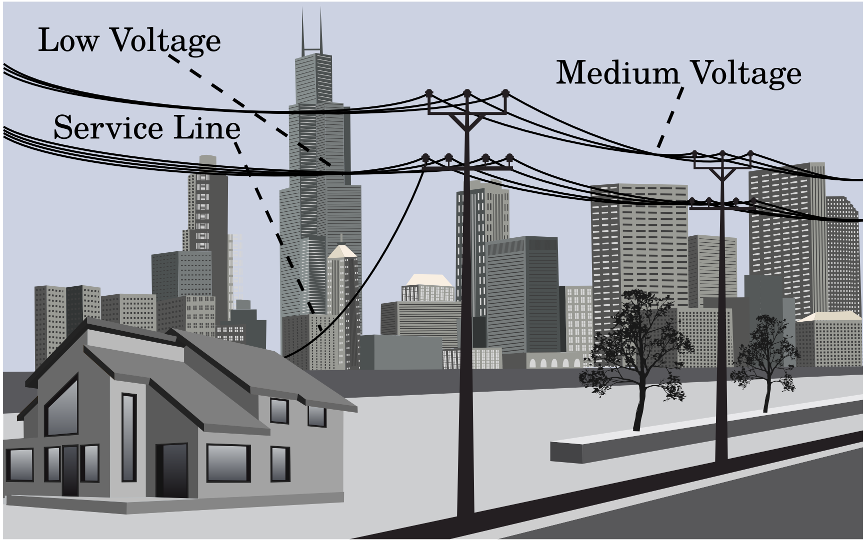


Figure 2.8: Distribution network.

distances are shorter and carry less power than the transmission lines. However, LV carries a greater current to transmit a given amount of power. There is also no need of big metallic poles to attach overhead lines, so the distribution poles are smaller and typically made of wood for the lower voltages (see Figure 2.8). The appearance of the distribution system is different depending on the region. These differences in design are consistent with differences in geography, population and load density, and the historical expansion of power systems (von Meier, 2006).

Another difference between the transmission and the distribution network is the topology. As shown in Section 2.1.2, transmission networks have a mesh network topology, while distribution networks usually have a radial topology or star network. In radial topologies, power flows only in one direction from the distribution substation to the particular load that requires it. However, with the increasing growth of new small local generators that feed electricity into the distribution power system, this assumption is not totally right. Nevertheless, the upstream and downstream of power flow is well separated, not as in transmission networks. The distribution lines leave the substations and spread in all directions to reach the users. Those lines and the components inside the network can only be energized from one direction due to its hierarchy. In order to protect the different circuits from the interruption or isolation of sections in the event of faults, it is necessary to take into account the direction of the power flows. In this type of topology, the design of the security measures is less difficult than in the mesh one. However, guaranteeing the security of supply is not as easy as in other topologies where redundancy between paths is available. Circuit breakers in radial systems can readily be located to isolate a fault quickly upstream of the problem. However, this causes the service interruption to all downstream components, without guaranteeing the supply.

In order to solve this problem, some distribution networks present a ring topology or loop topology, in which there are two power flow paths between the distribution substation and the load. In case of fault or maintenance of the system, the supply to the users is not affected. In order to isolate the paths, a circuit breaker is put in the redundant path and it is normally open. Therefore, it separates the ring into two radial lines, each one coming from different substations and under normal operation, the sections are not connected at that operating point. The switch can only be closed under certain circumstances, being one of the sections energized by the other. When this process is carried out automatically, it is often referred as “self-healing”. This topology also allows a transformer to pick additional load in case others are overloaded or out of service. Thus, the protection devices used in loop topologies have to be able to deal with power flowing in both directions through them. In highly dense urban settings, distribution networks also may have a mesh network topology, which may be operated as an active mesh network or a star network (MIT, 2011). Finally, the characteristics of the electricity supplied to customers are generally mandated by regulations. There are different requisites of the electricity supply, AC or DC supply, nominal voltage and tolerance, f, number of phases, maximum power instantaneously, $\cos(\Phi)$, earthing system, etc. In the next Section, the consumption of the different consumers will be detailed.

2.1.5 Electricity consumption

Finally, the last part of the electric system consists of consuming the generated and transported electricity. Electricity is consumed by a variety of loads, including lights, heaters, electronic equipment, appliances, pumps, etc. A load is defined in electric circuits as any device in which the power is being dissipated or consumed. Thus, consumption also includes the electricity consumed during its transmission and delivery. For example, the losses due to heating conductors or transformers step up or down voltage losses (Blume, 2007).

A load is defined by its impedance, which can be resistive, reactive or a combination of both of them (von Meier, 2006). Reactive loads can be divided also into inductive or capacitive loads. However, most of the loads are only purely resistive or a combination of resistive and inductive reactance. Resistive loads mainly consist of a conductor which is heated when the current flows through it, such as heaters or incandescent light bulbs. Inductive loads are the most common type of loads inside the grid. This type of load is in everything that has a coil, such as motors, fluorescent lights or transformers to condition the voltage. Capacitive reactance are part of electrical circuitry and they are characterized by storing energy inside them. Hence, they do not do any mechanical or other work. The impedance of an individual device may be fixed, as in the case of a simple light bulb, or it may vary, for example, if an appliance has several operating settings. Loads also differ in the type of electric power that they can use. Purely resistive loads only consume real power, they can be used on AC or DC, are tolerant to low voltage and indifferent to the direction of the current flow. Whereas loads with the presence of reactances typically only work on AC at a specific frequency and voltage. Inductive loads draw reactive power while capacitive loads supply it (von Meier, 2006).

Moreover, the relationship between these three type of loads influence system losses, revenues and reliability. In AC power systems, the relationship between v and i is influenced by them. For resistive loads a variation only in the amplitude between them is produced. However, for reactance loads apart from the reduction in amplitude, there is also another adverse effect, there is a time difference between v and i. This time difference is known as phase angle and can be of two types, for inductive loads i lags v and for capacitive loads i leads v. The phase difference affects to the amount of power that can do work, so it is related with the efficiency of the system. Thus, reducing the phase angle reduces the amount of i needed to get the same amount of work done in the loads. To maximize the efficiency of the system, the combination of the loads might be purely resistive to avoid the phase difference. The total power inside the system becomes real power and its total requirements and losses are minimum (Blume, 2007). Therefore, capacitors are connected close to large inductive loads to cancel their reactive power (i.e. increasing the $\cos(\Phi)$ of the load), reducing the burden on the network and the generators.

Customers think about their consumption individually, which from the grid perspective consists of numerous small and indistinct loads. However, it is very difficult for the operation of the grid to analyze the different load behaviors inside the power system. Thus, grid operators model the consumption aggregating loads, which consists of the customer combined effect in terms of magnitude and timing of electric demand (see Figure 2.9). The aggregation of loads can be done at different levels: an entire household, a building block or all the customers within a certain region. In addition, another difference between customers and operators is that customers think about the electricity in terms of energy while the operators refer to the instantaneous

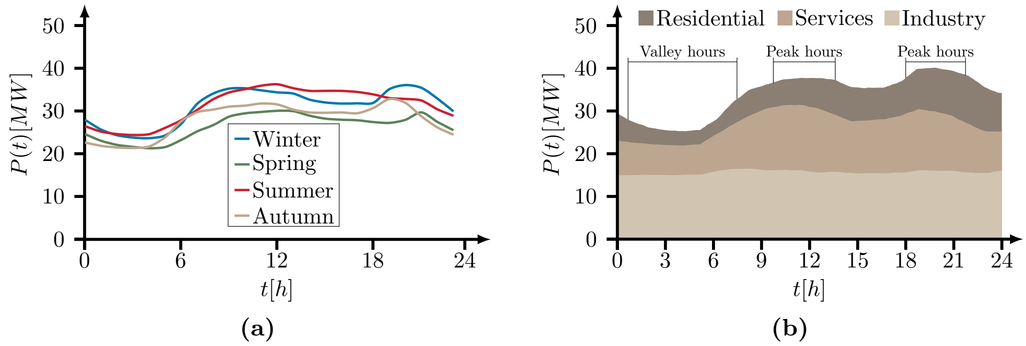


Figure 2.9: Spanish grid aggregated consumption: (a) variability of the consumption for different seasons and (b) residential, service and industry sectors. Source: Red Eléctrica de España.

rate of demanded power at any given time. Therefore, the demand refers to a quantity of power, not energy and is the key element on which the design and operation of the grid are based.

The operation of the grid must guarantee that the demand is met by supply anytime. Traditionally, the demand is considered as an independent variable in which users freely consume the amount of power they want and grid operators are in charge of bending over backwards if necessary to accommodate this demand. In this case, the demand is considered a variable beyond control, but recently this traditional point of view has changed. Nowadays, customers can vary their demand based on different strategies such as the electricity hourly pricing, the grid stability, better performance of the local resources, etc. This behavioral change of the users has been favored by emerging technologies that allow sensing, controlling and acting in different loads that compose the consumption. For example, new metering devices allow grid operators to know in real time the consumption of each individual and act on its consumption if the user allows them to control his demand. In addition, research and development efforts are directed toward technological approaches to make demand more responsive, including scenarios with remote-controlled and automated devices. Although these changes are making the grid evolving from a service-driven to a market-driven system, grid operators still consider the demand as a non controllable variable.

Thus, operators predict the status of the future demand in order to supply the right amount of electricity to all consumers. They use different intervals of time to do these forecasts, specifically they consider two periods of time, short term forecasts for a day or a week and long term forecasts for years. The data used to elaborate them is of different nature. The demand behavior is affected by numerous external factors, one of the principal data taken into account in the forecasts is the past demand behavior. The inter-daily variability (variations among different days) is very small for working days, however the difference is bigger between them and weekends (REE, 1998). Another important variable is the atmospheric weather. An example of the inter-seasonal variability is shown in Figure 2.9(a). In this Figure for the Spanish grid consumption, the variability and the shape of the aggregated consumption varies depending on the season of the year. In winter, the maximum value of the demand is reached around midday, due to working hours and the fact that at homes where the lunch is cooked. In addition, there is another maximum around 20 h because it is the time when people are reaching home. During summer, the demand drop between the two maximums disappears due to the air conditioning. The demand is greater in winter and summer comparing to the demand of autumn and spring due to weather conditions which affect less to the consumer behavior and they do not need any air conditioning device. There are other random factors that affect the behavior of the demand, such as football matches, school holidays, new technologies, etc.

The reason behind demand forecasting and other methods implemented by grid operators is the security of supply since the grid operates in real time. As load increases, generation must increase to supply the consumption with the appropriate v and f. Otherwise, v would collapse and f would drop. In addition, the losses also increase with the consumption increase. Thereby, knowledge about future demanded power in an electrical system is essential in order to anticipate and supply the right amount of power at every moment (Blume, 2007).

In the utility context, two other concepts are used to describe the demand, the coincident and noncoincident demand. Coincident demand refers to the amount of combined power demand that could normally be expected from a given set of customers. Whereas, the noncoincident demand is the total power that would be demanded by the costumers if all their loads were operating at the same time. However, these two concepts do not normally coincide. Thus, coincident demand reflects the statistical expectation about how many loads will overlap at any time. It is a prediction that becomes more reliable when higher number of customers are involved. Although ordinarily the grid operator observes only the coincident demand, it must be prepared to face noncoincident demand under certain circumstances. The reason why noncoincident demand is undesirable is because if all the loads begin to operate at the same time, the inrush current as they turn on is too high.

Analysts of the grid use different representations of the aggregated consumption to understand better its behavior along a period of time. Instantaneous demand varies inter-daily and intra-daily, which means that it varies over the course of a day. A load profile is used to represent the demand during a day, or a statistical average over typical days for a given month or season. It consists of representing the amount of power consumed at each moment during a day. The load profile can represent different levels of aggregation: individual electricity users, distribution power line or an entire grid. In Figure 2.9[2], the Spanish consumption is represented for a 24 h load profile. There exist different periods around the day that are very interesting for the grid service providers to adjust the generation supply. In particular, there are two troublesome periods: the periods in which the demand is maximum, which is called peak demand, and the periods in which the demand is minimum, which is called demand valley. In Figure 2.9(b), there are two peak periods of time, one at midday and the other one at the evening, while there are another two valley periods, one between the two peak periods and the other one at night, typically of a winter day (REE, 1998). However, in summer the peak period occurs in the central hours of the day, coinciding with the hours of the highest temperature (see Figure 2.9(a)). During peak periods, it is more expensive to produce electricity because it is necessary to operate plants whose production is more expensive and produce more CO_2, such as combined cycle plants. But, the power system has to be sized to meet this demand in spite of being for a few hours or even minutes. It is also important to know when the valleys are produced in order to reduce the generation and adapt to the demand.

From the power system perspective, it is relevant to compare periods of higher and lower demand over the course of a year. For example, a measure to know the difference between seasonal maximum power consists of comparing the maximum power of each month of the year. This measure shows how the peak load evolves during the year and also offers a comparison between seasonal and daily rhythm. In warmer climates where air conditioning dominates electric usage, demand will tend to be summer-peaking; conversely, heating-dominated regions will see winter-peaking demand (von Meier, 2006).

Another useful representation of the demand, different from the load profile, is the load duration curve (see Figure 2.10). The load duration curve still depicts instantaneous demand over certain time periods (generally in hour intervals). However, the time axis is not sorted temporally, it is sorted according to the demand in each hour, from the maximum to minimum demand along the year. Thus, the highest demand hour of the year appears at the first hour, continuing in a monotonically decreasing fashion. The load duration curve provides a useful representation of how

[2]The load profile of the Spanish grid consumption is available at REE, monthly data are available since 1995. Moreover, the European Network of Transmission System Operators (ENTSO-E) data portal has monthly data about consumption and production for different European countries.

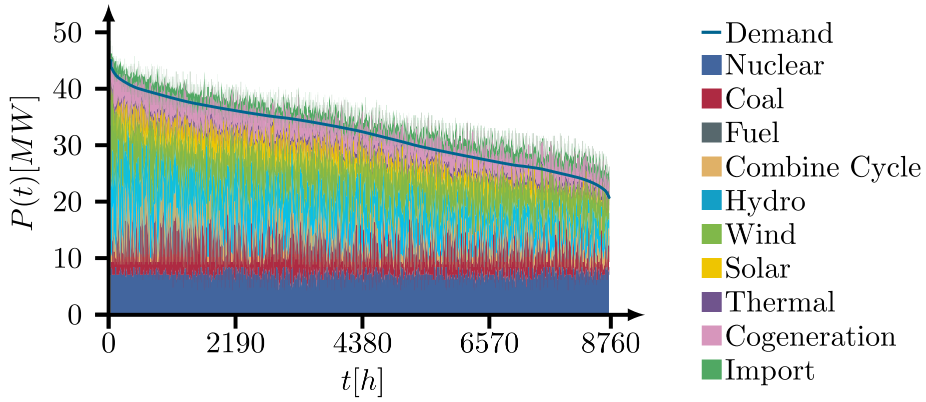


Figure 2.10: Load duration curve plus generated power for the Spanish grid for the 8760 h of 2015. Source: Red Eléctrica de España.

the load varies and for how many hours in a year it is above a particular level. Figure 2.10 shows an example of the load duration curve, in which the 8760 h of the year are represented. The highest demand is at the left-hand side corresponding in general to the peak hours of the year, while the lowest demand is at the right-hand side corresponding to the night hours mostly. In Figure 2.10, generated power is also represented for the different sources available in the Spanish power system. It can be observed how much electricity of each form of generation has been supplied to the demand. This curve type of Figure 2.10 is a useful way to characterize the pattern of demand, because it is easy to identify the maximum of the demand and its needs.

The higher the slope of the load duration curve is, the bigger is the difference between the peak hour and the valley hour of the year. In addition, it is more expensive to meet the needs of a spiked load duration curve than a flat one. The reason is that the generation capacity needs to meet the peak load to guarantee the supply, while the utilization of the generation is related to the average load. So, the cost of providing the service is large and is highly related to peak capacity required by the power system. A pronounced peak indicates that grid oversizing is required to meet demand on just a few occasions. Thus, the assets required to supply the peak do not tend to be used the remainder of the year (von Meier, 2006). For example, in Spain the 300 h of greater consumption account for 10 % of the annual generation, around 4 GW. Therefore, the system must be designed to meet that demand, and the investment done to meet those hours is large compared to the time of use.

There is a metric used to measure quantitatively the flatness of the load duration curve. This useful metric is called load factor (L_f) which is defined as the ratio between average demand and peak over a certain time period. The value of L_f characterizes how flat the demand is. If $L_f = 1$, that means that the demand is flat. However, this is an ideal value because the demand present peaks of activity. Normally, $L_f < 1$ indicating how big the peak of the demand is in the time period evaluated. The bigger the L_f value is, the higher the peak of the demand and the slope of the curve are. For example, for the Spanish consumption in 2014, according to Red Eléctrica de España (REE), $L_f = 0.72$.

The L_f ratio obviously depends on climate and the other factors, but it also depends on the load diversity within the customers. Traditionally, the L_f enhancement has been performed by introducing greater diversity of demand, so that the resources are shared for meeting the peak. For example, commercial loads that operate during the day are complemented by residential loads before and after work hours. However, nowadays it is difficult to incorporate more diversity in the grid. Thus, Demand Side Management (DSM) was born to minimize the need for additional supply as the demand is reduced or controlled. DSM programs are designed to provide assistance to consumers in order to help to reduce their energy demand and control their energy cost while delaying the construction of generation, transmission and distribution facilities (Blume, 2007).

For a better understanding of the demand composition, the grid operators tend to group the customers types inside the grid. This classification groups users by activity sectors and consists of dividing the demand in three main groups (see Figure 2.9(b)):

i) Industrial. The industrial electric consumption is a baseload consumption and it appears to be constant during the whole day. According to the International Energy Agency (IEA) in 2012, the electricity consumed by industrial loads was a 50 % of the world electricity. In the industry the electricity is used as the driving source of large electric motors and special machines of each sector. It is also used to heat the contents of tanks, reservoirs or boilers. However, industrial consumption has a large reactive component (Q) due to the AC motors used. Thus, they need capacitor banks to counteract the inductive loads, improving their $\cos(\Phi)$. Depending on the size of the industrial consumer, they can be connected to MV or LV distribution networks. The larger industry consumers (i.e., military bases, oil refineries, mining industry, etc.) normally have their own substations facilities for conditioning the power to their necessities. These facilities include lines, electrical protection equipment and transformers to step down or up the voltage to supply their electrical needs.

ii) Services. In the service sector, their load profile present peaks of demand coinciding with the working hours of these establishments. The periods at which they are open depends on the country. However, the rest of the time they maintain their consumption almost constant. According to the IEA in 2012, the electricity consumed by service loads was a 25 % of the world electricity. This sector consists of heterogeneous activities, which include mercantile and service, office operations, warehousing and storage, education, public assembly, lodging, health care, and food sales and services. Among the different loads that are part of the service load profile includes larger-scale lighting, heating, air conditioning, kitchen apparatus, and motor loads such as elevators and large clothes handling equipment. So there exist a base consumption to feed those devices that has to be on in spite of finishing their daily activity. Typically, the services loads work at LV.

iii) Residential. Finally, the residential customers present peaks and valleys periods of time. Their consumption is lower than in the other two sectors, but its variability is bigger. According to the IEA in 2012, the electricity consumed by the residential sector accounts for approximately 20 % of the world electricity. The larger use of electrical energy is mainly driven by the following devices in a highly electrified home: air conditioning units, refrigerators, stoves, space heating, electric water heaters, clothes dryers and washing machines. However, there are other devices that also consume electricity to a lesser degree, such as lighting and consumer electronics (TVs, radios, personal computers). The peaks of the residential demand match with the morning, lunch and evening parts of the day, that it is when people are normally at home. The rest of the time the residential load profile consists of non-stoppable appliances and stand-by devices. Typically, the residential loads work at LV.

A sector that is becoming particularly relevant in electricity consumption is the transportation sector. According to the IEA data for 2012, the electricity consumed in this sector supposes around 1 % of the world electricity. The transportation methods that use electricity include the tram, subway or train. Nowadays, there is a special interest to use electricity in order to replace the fossil fuels that propulse most of the vehicles around the world. With the use of the Electric Vehicle (EV), important economy improvements for road transport are being developed. A growing interest in EV is being conducted by diversifying energy sources for the transportation sector and it is influencing vehicle technologies (IEA, 2014c). EV can be classified in different types: electric as the only source of energy, EV, with electric and fossil fuels engines but with no plug-in to recharge the battery, Hybrid Electric Vehicle (HEV), and

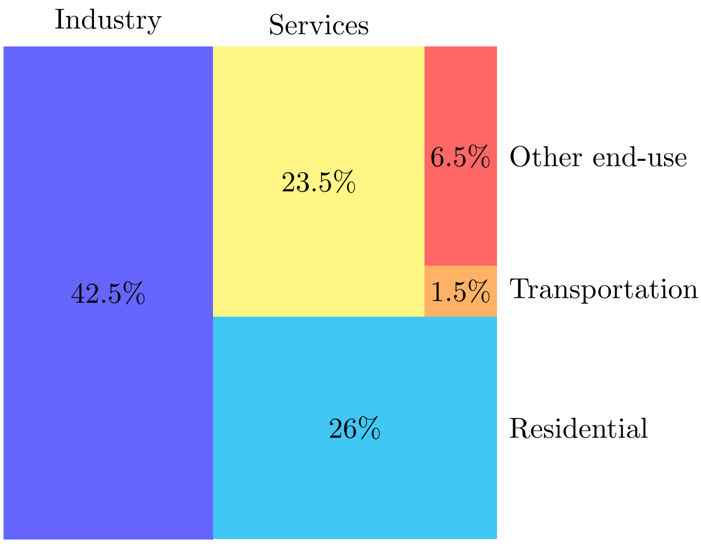


Figure 2.11: Electricity use by sector of occupation.

plug-in vehicles driven by electricity and fossil fuels, Plug-in Hybrid Electric Vehicles (PHEV). The development of infrastructures is as important as the development of EV transportation technology. This market is continuing growing, but slowly. In 2013, HEVs reached 1.6 % of global market share, while EVs were around 0.4 % of the global market share of vehicles (IEA, 2014c).

The electricity consumed by each sector can be observed in Figure 2.11. The main consumer of electricity is the industry sector with a 42.5 %, followed by the residential sector with a 26 % and then the services sector with a 23.5 %. Those are the main sectors of activity related with using electricity. However, other sector are growing its part of the share such as the transportation sector which still represents a low percentage of the total with only a 1.5 %. This figure will be increased in the next few years as the technology of EV matures. EV plays an important role inside the future of the SG as a mobile load that can be connected in different parts of the power system. The world electricity consumption represented a 18.1 % of the total energy consumption and the trend of consumption is positive (IEA, 2014d). However, the resources to feed the global electricity needs are limited and the demand could not be supplied in the future at the actual growing rate. So, new solutions for better use of available resources should be conducted, such as the SG.

2.2 Grid needs

With the description of Section 2.1 of the different grid parts, a better understanding of power system is achieved. However, grid operation is not trivial and embodies different challenges. If those challenges are not handled effectively, power systems cannot fulfill their objective to supply electricity to the demand. Thus, power systems are operated from a certain level of centralization to guarantee the security of supply.

Grid operation is carried out at control centers that perform three main functions: i) monitoring, ii) analysis and iii) control of grid status. Monitoring consists of supervising the grid for proper operation, warning when any fails occurs. The raw data, received at control centers, are analyzed to give the insight of the current and future grid states. This suite of tools is known as energy management system. Finally, the control centers calculate the expected hourly power of generating units for a time period in the future, normally a day ahead based on the demand forecast. Then, this information is passed to the different generator units. The decision of which units should be generating for the next day is known as unit commitment and the specification of the output power of each generator is the economic dispatch. Depending on the area, the unit commitment and economic dispatch values are calculated based on external factors such as fuel costs or wholesale markets. These estimations from control centers are continuously updated to minimize the total costs given the load level, generator availability and transmission constraints. Control centers also manage to match small changes in load by adjusting generator units and meet the scheduled power exchanges with neighboring systems. This control mechanism is called automatic generation control (MIT, 2011).

Grid operation is a complex process in which several factors have to be taken into account. It faces countless problems which have to be solved in the less time possible to ensure that generation meets demand. However, electric power industry stakeholders (utilities, vendors, manufacturers, regulators, consumers and their advocates, and governments) have identified some issues that motivate a deep change inside the grid. These changes are encouraged by different factors such as new emergence technologies that help to increase the grid efficiency. The current electrical grid was designed to operate as a vertical structure consisting of generation, transmission, and distribution and supported with controls and devices to maintain reliability, stability, and efficiency (Momoh, 2012). This structure has barely changed since its inception, although new advances have been incorporated to help their operation and management. These changes are not enough to face the current needs of the different electric power industry stakeholders. Some identified issues must be solved in order to improve the current system and evolve to the next grid generation. The principal problems are described below (El-hawary, 2014; Momoh, 2012; Ekanayake et al., 2012):

- *Ageing infrastructure.* In a lot of countries around the world, grids were deployed more than 50 years ago when they suffered a rapid expansion to cover their whole national territory. Part of the devices from transmission and distribution networks are old and need replacement. The costs to introduce new equipment are too high and grid operators cannot afford the investment necessary to replace the old ones. However, the replacement is needed and it is a good opportunity to improve grid design and infrastructure according to current energy needs.

- *Electricity growth.* Electricity is the final energy form more used at the present. Almost all devices used daily are powered by it, such as smart phones, computers, appliances, lights, etc. Demand will continue growing due to different factors, such as the development of countries, coverage of new necessities, etc. Electrical dependency is notable in developed countries and is growing in developing countries. Electricity is set to remain as the fastest-growing final form of energy worldwide. World electricity demand is expected to grow by 2.1 % per year on average over 2012-2040 with its share from the rise in total energy use for all sectors and regions (IEA, 2014d). Figure 2.12(a) shows the electricity demand by regions in 2012 which is smaller for developed countries and higher for developing and underdeveloped regions. The growth of electricity demand is exponential, as it can be observed in Figure 2.12(a). China and India will consume more than twice the current electricity demand by 2040. In contrast, the OECD countries have a lower growth due to energy saving policies and the saturation of their electrical systems. Thus, it is necessary that the grids are prepared to absorb this growth and supply the new demand. However, for the current structure of the grids, it is difficult to meet this new demand. The reason is that the grids are operating at their capacity limit and the introduction of new demand will make the security of supply difficult. It is a great opportunity to invest and expand the network in both size and functionality.

- *Efficiency.* The grid is a huge extensive network which presents losses in all its constituent parts despite ensuring electricity supply. However, its efficiency is growing as a result of countless routine actions, although it is not enough to tackle all the losses involved in the entire process. The largest losses inside

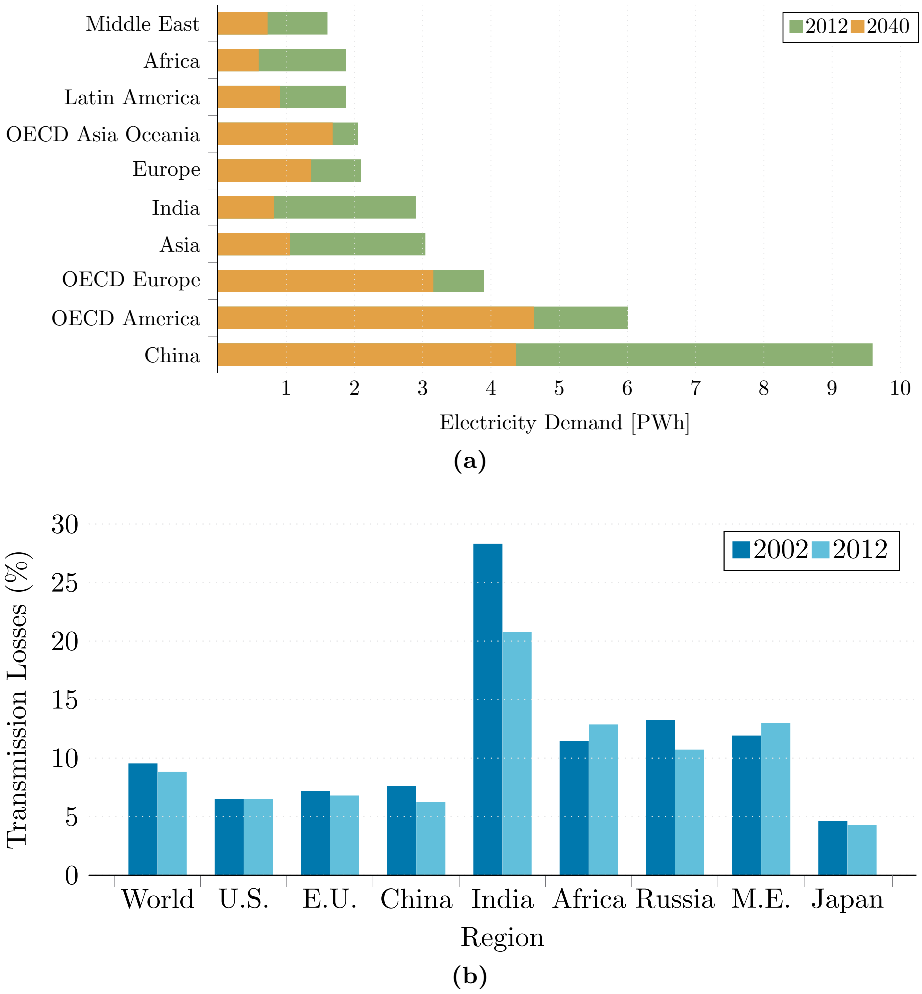


Figure 2.12: World statistics: (a) electricity demand growth rate and (b) transmission and distribution losses rates. Source: International Energy Agency.

the grid are due to the transportation and distribution of electricity. These losses include those occurring in the equipment of substations (transformers, circuit breakers, etc.), power lines, etc. Figure 2.12(b) shows the rates of losses in the transmission and distribution of electricity in 2002 and 2012. In 2012, these losses represented the 8.8 % of the world total generation. The transmission losses depend on the distances that electricity has to cover and how the population density of the country is. Thus, Japan counts only with less than 5 % while Russia is above 10 %. The grid would be more efficient if generation was closer to consumption. The savings will be not only achieved in the efficiency of the grid, but also in the investments required to deploy large transmission and distribution network. However, most of the losses occur in the generation side. The reason is that the transformation process of raw materials to produce electricity is inefficient and the equipment to do it is not extracting all the potential from them. These losses represent 62.4 % of the world electricity production (see Figure 2.13). The only way to reduce them is to improve the conversion efficiency and the equipment part of the process. In addition, new generation methods can reduce these losses, such as renewable energies whose raw materials are infinite. Conversion losses can also decrease its value if demand uses more efficient equipment.There exist lots of energy efficiency policies to enhance the use of more efficient equipment in the demand

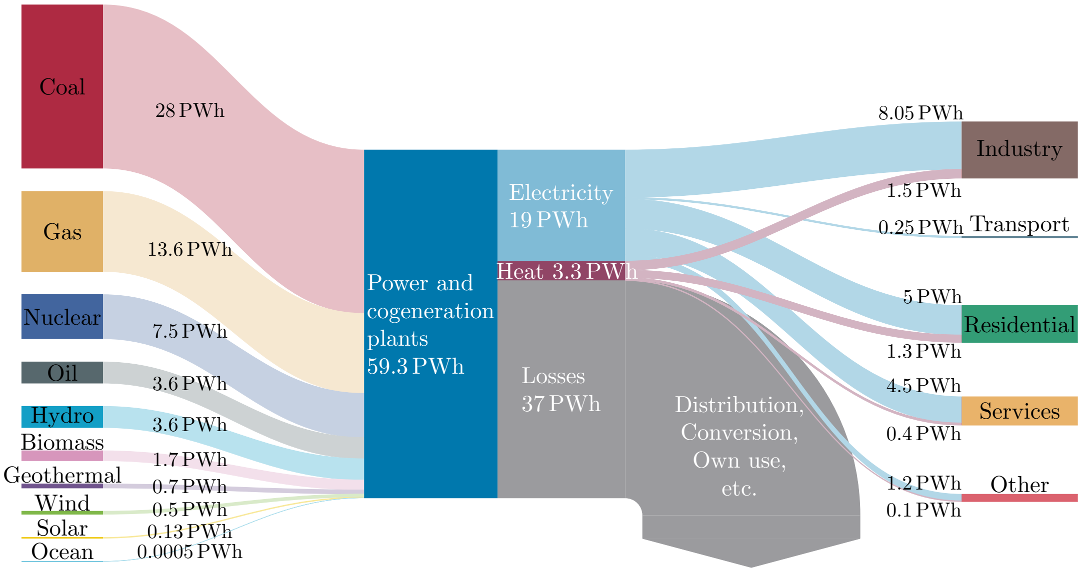


Figure 2.13: Energy transformation diagram from generation to consumption for the year 2012. Source: International Energy Agency.

side. In this way, the generated electricity will be lower and the losses will be reduced.

- *Generation paradigm.* The generation mix of a power system consists of the different electricity generation forms that compose the system. It is currently dominated by large power plants far from the places where the generated electricity is consumed. These power plants might be excellent economically, but the electricity is transmitted over very long distances and the energy and environmental performance is low, as seen before. They are located in certain places far from the consumption depending on economic, security, logistics and environmental factors. The main generation type in the world electricity mix is occupied by thermal power plants. The vast majority of thermal power plants used as primary energy source fossil fuels (coal, oil and gas, see Figure 2.13). World electricity production from fossil fuels represents around 75 % and almost half of world electricity is still produced by burning coal, a major contributor to global warming. Burning fossil fuels produce large amounts of CO_2 emissions which are harmful and promote the greenhouse effect, increasing the temperature on Earth. Thus, public interest groups are putting pressure on politicians to reduce CO_2 emissions through the adoption of alternative energy sources and regulations to increase energy efficiency. It is in this scenario where Distributed Generation (DG) arrives to solve some issues around traditional ways of generating electricity. Investments are done to support the growth of these technologies and change the mix of power generation in which central power plants operate in parallel with large numbers of small and decentralized generators. Figure 2.14 shows the distribution of the different power plants in Spain. The generation mix of Spain consists of large generators plus DG (solar and wind power plants). The traditional generators are easily located in the map while DG ones are agglutinated by region because extensive generators are located throughout the entire region. One of the DG features is that it is generally close to the demand, saving the transportation losses. Another advantage is that they produce less CO_2 emissions due to the nature of raw materials (wind, sun, water, etc.). On the other hand, the intermittent and fluctuating energy availability of renewable energy sources, such as wind and solar, make their integration difficult inside the generation mix of the current grids. Besides the current state of the grids, the lack of communication between

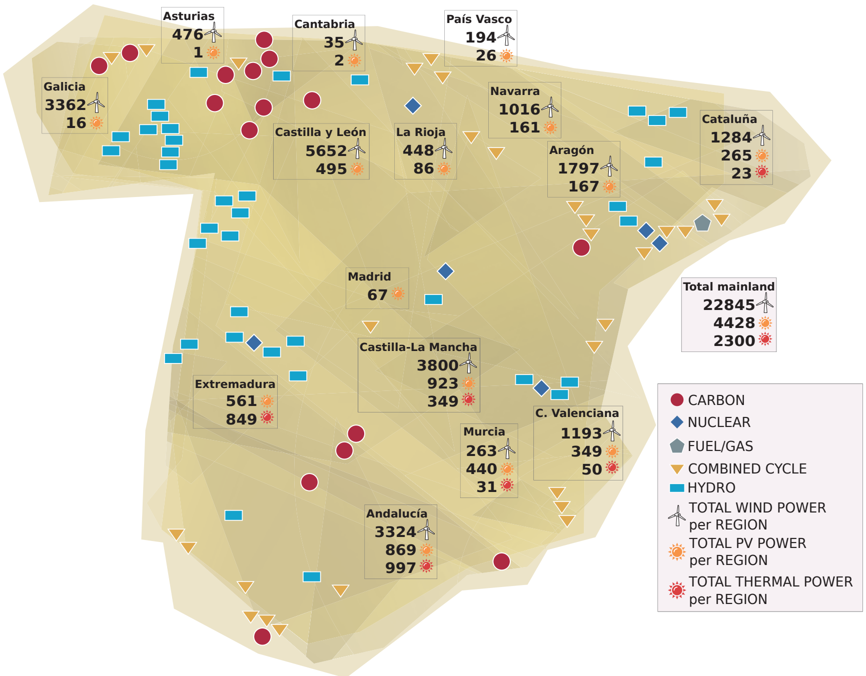


Figure 2.14: Map of the generation power plants of the Spanish grid. Source: Red Eléctrica de España.

DG and the grid is reaching their capacity limits. DG fluctuations must be counter-balanced with more intelligence in the grid and storage.

- *Grid operation.* At the operation level, different constraints are identified that make difficult the daily operation. One of them is provoked by thermal constraints produced in transmission and distribution lines and the equipment affecting their power transfer capability. Thermal constraints depend on environmental conditions, that change through the year. Grid operators are constantly monitoring the variables that compose the electricity signal. v and f are required to be inside their limits to guarantee the proper functionality of the electricity devices. Low voltage levels may cause malfunctions of customer equipment and excess of i inside the transmission and distribution network. f is strictly maintained between its limits, because f is affected by the balance of generation and demand. Any imbalance is reflected as a deviation in the f. To maintain the system stability, grid operators use response and reserve services to bring the f back within its operating limits. As a last resource, some loads are disconnected from the grid to prevent major faults. Nowadays, the daily grid operation is becoming more difficult as more generation is being connected to the distribution network (DG). If local generation is not coordinated, the DG can cause over-voltages. Thus, maintaining the generation-demand balance and the system f within limits become difficult. In addition, another element that makes difficult the daily grid operation is the variability of the demand, throughout the day and across seasons. The grid has to supply the electricity to the corresponding demand at each instant. Hence, the electricity system infrastructure is designed to meet the highest level of demand, so during non-peak times the system is typically underutilised. The economic investment to satisfy this condition is too high that would not be needed if the demand curve was flatter. Traditionally, grid operators try to smooth the aggregated consumption adding diversity in the demand. Nowadays, the problem resides in the communication from the grid to the user that it is almost nonexistent and the lack of controllable loads to shed them from the grid. Moreover, consumers are requiring more transparency in the consumption and new pricing models to grid operators. This lack of information in the distribution side is not only affecting consumers, but also to the appearance of new products, services and markets and the accommodation of all DG and storage options, such as the EV. It is thought that in future the electrification of domestic heating loads and the EV charge will lead to a greater capacity of flexible loads. This would help to maintain the network stability, reduce the requirements for reserve power from part-loaded generators and the need for network reinforcement. Therefore, utilities need to adopt information and communication technologies to handle new operational scenarios and challenges while maintaining profitability and investing in infrastructure.

- *Reliability and sustainability.* Electricity security is a matter of keeping electricity available to all connected users, within acceptable standards and the amounts desired at any given time. At present, consumers require an increasingly reliable electricity supply as more and more critical loads are connected. The traditional approach to improve reliability was based on the installation of additional redundant circuits, at considerable capital cost and environmental impact. Other measures consist of disconnecting the faulty circuit, so no action was required to maintain supply after a fault. However, recent system failures and blackouts have focused attention on trying to increase system reliability. Reliability could be defined in terms of two basic and functional aspects: adequacy and security. The system adequacy is the capability of the power system to meet changes in aggregating power requirements, using existing and new resources. On the other hand, the system security consists of maintaining the supply even with unexpected surges or sudden disruptions in demand. Accomplishing both features require efficiency, time and investment. In spite of the system fast response to changes in demand, it is not enough when the variability is too high. That is one reason to interconnect the power systems of different regions, in order to guarantee the supply. In addition, when a fault occurred and a blackout is produced, the reactions of grid operators are fast but not enough. The demand growth makes necessary to increase the response to changes in the demand as well as the need for more generation to cover this increase. However, the current flexibility of the power systems is not optimized since its capability to alter the consumption or generation in a rapid and large imbalance is relatively low. For a greater adequacy, numerous mechanisms can be introduced, such as the electricity trading, storage systems or more automation in the demand. About the security of the power systems, the ability of the grid for self-healing (anticipating responses to system disturbances) is very low. They only respond to prevent further damage and are focused on protecting assets following a fault. Efficient and reliable electricity transport are fundamental to maintain functioning economies and societies. By watching these system features and increasing their efficiency, grid sustainability could also be incremented as CO_2 emissions decrease and more renewable energies are integrated.

- *Investment and costs.* The global generation costs are rising their figures as demand is also growing. Generation investments are required to meet the demand. Traditionally the investments were done in large power plants which can generate large amounts of electricity. However, nowadays these investments are being transferred to smaller generators that are closer to the consumption and use renewable energy. Different reasons support these change of investments. One of them is the raising prices of raw primary materials, especially fossil fuels because reserves are dwindling. But also regulatory pressures are changing the model of investment, such as regulations in favor of the CO_2 emission reductions or the improvement of electricity savings. Special efforts are being done by investors to afford new models to optimize assets and operate efficiently. The investments are recovered through regulated or competitive wholesale electricity prices and, to a lesser extent, supportive measurements from the governments. New markets are being developed, such as electrification of transport, to favor a major energy trading and expand the use of electricity. This results in a power system more complex to operate and the necessity to adequate it to these new products and services. Thus, high investments have been done in the infrastructure (transmission and distribution networks) to expand the grid, meeting higher demand and stability. Success in financing all the needed investments depends largely on governmental policies and regulatory frameworks. But governments should reduce the political and regulatory uncertainties in order to generate a market attractive for investors. Combined with the increasing demand, higher prices are set to drive up electricity bills. The price that a consumer pays for electricity has several components, some of them are: wholesale price to cover generation cost plus a margin, cost of transmitting and distributing the electricity, retail costs and taxes.There are high differences in electricity prices depending on the region. Therefore, regulators are pushing for more competitive and lower energy prices. Also, improvements in energy efficiency could substantially moderate the impact of rising prices on electricity bills.

It is necessary that the grid improves its behavior to continue supplying electricity to consumers in terms of quality and security. In addition, if those needs are not tackled in the near future, the power system will continue ageing without being able to incorporate new opportunities as they arise into the market scene. Improving the reliability, availability, and efficiency of the grid would benefit all members of the grid, reduce the costs and investments in the system and give the grid more responsiveness and flexibility. Apart from the economic and policy motivations, a proper grid development would also contribute to the advance in emerging technologies, such as communications, computing power, energy storage, and renewable generation. These new advances would facilitate new services and improve monitoring, control, communication and self-healing technologies. Moreover, making consumers participant in the grid structure will optimize the operation of the system and provide them with information and choice of supply. The promise of a smarter grid is done in which all the actual concerns are solved and the quality of the system is improved to advance towards the future.

2.3 Grids evolution: towards the Smart Grid

As explained in Section 2.2, the grid has recently faced important challenges to secure electricity supply in a more efficient way according to more restrictive regulations. The advance of new technologies has helped to alleviate some problems in the past. However, there is a pressing need to accelerate the grid development that addresses the global challenges of energy security, climate change and economic growth with advanced low-carbon energy technologies. It is in this scenario where a new concept of grid arises known as Smart Grid (SG). The SG also receives different names, smart electrical/power grid, intelligent grid, intelligrid, futuregrid, intergrid, or intragrid, but they all consist of enhancing the 20th century power system (Fang et al., 2012). The "smartening" of the grid is an evolutionary process that involves adding new elements at the same time that the communication infrastructure is improved to manage each part of the grid. This evolution is shown in Figure 2.15, in which differences between the past, present and expected future of the grid are appreciated. It is fundamental not only a higher integration of every system part, but also to facilitate the entry of new agents and technologies inside the grid. As shown in Figure 2.15, over the years the grid has been responsible for expanding the communication network between its parts. However, it remains to cover the last few kilometers related to the users. Although many regions have already begun to "smarten" their electricity systems, all regions will require significant additional investment and planning to achieve a smarter grid (IEA, 2011).

There is not a specific definition of what a SG must be. The reason for not adopting only one concept is due to the diversity of power systems around the world, in spite of sharing some features in common. There are differences mainly due to the

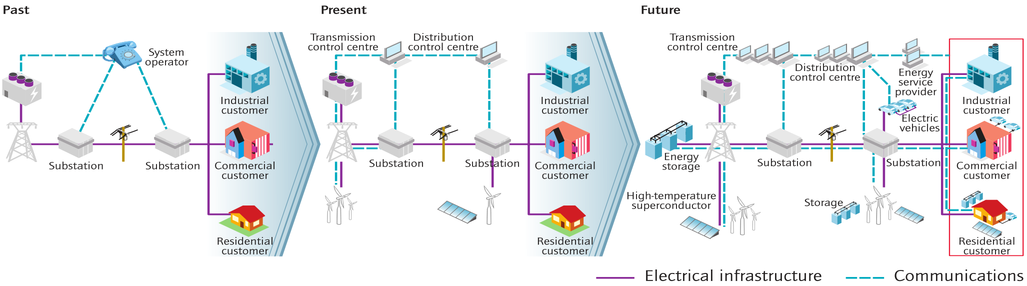


Figure 2.15: Evolution of the grid structure. Source: IEA

location and construction adopted by each government of the countries where they are located. Those governments are encouraging SG initiatives as a cost-effective way to modernise their power system infrastructure. The idea of the SG was conceived as an Advanced Metering Infrastructure (AMI) to improve the demand response and energy efficiency with self-healing mechanisms inside the grid to avoid sabotages and natural disasters (Rahimi and Ipakchi, 2010). However, this definition is also evolving with time as new requirements drove from the different grid members. There are different definitions made from a variety of organizations to understand what a SG is. All the definitions have some concepts in common that combine new technologies, end-user solutions and address different policies and regulations. For example, the European Technology Platform defines the SG as (European Commission, 2006):

> *"A SmartGrid is an electricity network that can intelligently integrate the actions of all users connected to it – generators, consumers and those that do both – in order to efficiently deliver sustainable, economic and secure electricity supplies."*

On the other hand, SG is defined in the U.S. Department of Energy (2009) as:

> *"A smart grid uses digital technology to improve reliability, security, and efficiency (both economic and energy) of the electric system from large generation, through the delivery systems to electricity consumers and a growing number of distributed-generation and storage resources."*

While in the UK Department of Energy & Climate Change (2009), the SG is identified as:

> *"A smart grid uses sensing, embedded processing and digital communications to enable the electricity grid to be observable (able to be measured and visualised), controllable (able to manipulated and optimised), automated (able to adapt and self-heal), fully integrated (fully interoperable with existing systems and with the capacity to incorporate a diverse set of energy sources)."*

In Momoh (2012), it is suggested that the SG must be:

> *"The smart grid is an advanced digital two-way power flow power system capable of self-healing, and adaptive, resilient, and sustainable, with foresight for prediction under different uncertainties. It is equipped for interoperability with present and future standards of components, devices, and systems that are cyber-secured against malicious attack."*

Last but no least, the Canadian Electricity Association has defined it as (Canadian Electricity Association, 2010):

> *"The smart grid is a suite of information-based applications made possible by increased automation of the electricity grid, as well as the underlying automation itself; this suite of technologies integrates the behaviour and actions of all connected supplies and loads through dispersed communication capabilities to deliver sustainable, economic and secure power supplies."*

There are other definitions besides those already mentioned here. They present the SG as an opportunity to evolve the actual status of the grid and improve its operation. Some people believe that SGs are nothing more than a step forward to modernize a "dumb network". However, this assessment is totally wrong because the "smartening" of the grid consists of interconnecting all the parts to have control of each one in real time for better utilization of resources. In this Thesis, a broader view of the term and what the SG involves are addressed because the term SG means different things to different people and its meanings are continuously evolving.

So far, the global vision of the SG gathers together the different parts of the current grid and adds different communication technologies with computational capabilities to provide favorable attributes that enhance its operation (Gharavi and Ghafurian, 2011). A grid to become smart must include the following attributes among others (Fang et al., 2012; El-hawary, 2014; IEA, 2011):

- *Optimised and efficient.* The SG incorporates the latest technologies to optimise the use of its assets, in order to meet the increasing demand without adding extra infrastructure. Dynamic ratings are used to optimise the capacity of the system, allowing greater demand by continuously monitoring and rating their capacities. Real-time monitoring and system control allow increasing the equipment efficiency through condition-based maintenance, signalling the right instant in which equipment maintenance is needed. Moreover, this platform will allow a smarter operation of the delivery system, rerouting power and working autonomously, pursuing efficient asset management. Monitoring and controlling the assets of the power systems allow reducing losses and eliminate congestion, including utilising them depending on what is needed and when it is needed to increase the efficiency of the system. In addition, operating efficiency increases when selecting the least-cost energy-delivery system available through these types of system-control devices. Giving real-time consumption measures to users will provide them with the information needed to modify their consumption profile based on individual preferences (pricing, efficiency, etc.) to enhance the operation of the power system.

- *Integration.* One of the virtues of the SG is the capability to integrate inside the power system any generation technology without wasting electricity. It accommodates not only large centralised power plants, but also the growing array of customer-sited distributed energy resources. This integration covers different technologies including renewable, small-scale combined heat and power, and energy storage, which will increase rapidly all along the value chain, from suppliers to marketers to customers. Thus, the SG will provide simplified interconnection similar to "plug-and-play".

- *Participation.* Enabling two way communication through different parts of the power system would help to decrease the peak-valley difference in the daily demand. The reason is that consumers help to balance supply and demand, ensuring reliability by modifying the way they use and purchase electricity. These modifications come as a result of consumers having choices that motivate different purchasing patterns and behaviors. These choices involve new technologies, new information about their electricity use, and new forms of electricity pricing and incentives.

- *New horizons.* The SG creates new opportunities and markets for its different participants. The efficient operation of the markets gives users the opportunity to choose among competing services. SG opens access to markets through increased transmission paths, aggregated supply and demand response initiatives, and ancillary service provisions, which are those services that facilitate and support the continuous flow of electricity. In order to achieve the incorporation of these new markets, power system operators must manage some different variables such as energy, capacity, location, time, rate of change and quality. Regulators, owners/operators and consumers need the flexibility to modify the business rules in order to suit operating and market conditions.

- *Power quality*. The SG is able to supply varying power grades and prices to all the different consumers that are inside the power system. It is able to supply the power quality necessary that meets the different needs of consumers, free of sags, spikes, disturbances and interruptions to power the different loads of the digital economy. Not all commercial enterprises, and certainly not all residential customers, need the same quality of power. The cost of premium power-quality features can be included in the electrical service contract. Advanced control methods monitor essential components, enabling rapid diagnosis and solutions to events that impact power quality, such as lightning, switching surges, line faults and harmonic sources.

- *Resilient*. The SG assures and improves the reliability and security of supply as it becomes more decentralized and reinforced with security protocols. Resilience of the SG refers to the ability of a system to react to unexpected events (disturbances, attacks, natural disasters, etc.) by isolating problematic elements while the rest of the system is restored to normal operation. Some protective measures are predictive maintenance, self-healing and strengthening the security of supply through enhanced transfer capabilities. These self-healing actions result in reduced service interruptions to consumers and help service providers to better manage the delivery infrastructure.

- *Sustainable*. The advance of global climate change is slowed down thanks to a high integration of low carbon technologies inside the SG. The reduction of fossil fuels, through new ways of generation and the use of EV, is favoring a path towards significant environmental improvements. In addition, the reduction in demand variability will improve the operation of the system and a better operation that affects positively the environment. Using local renewable generators will decrease the environmental impact in reducing both CO_2 emissions and relevant visual effects of large power plants.

Thus, a possible definition of the SG that includes all these concepts is suggested:

> *A SG is an electricity network that uses communication technologies to co-ordinate the needs and capabilities of all generators, grid operators, end-users and electricity market stakeholders to operate all parts of the system as efficiently as possible, minimising overall costs and environmental impacts while maximising system reliability, resilience and stability.*

In spite of the benefits of the upgraded grid, there are also some impediments and drawbacks that make difficult its adoption. These concerns are related with the different parts that shaped the SG and its deployment. Various barriers have been identified and are described below (El-hawary, 2014; Luthra et al., 2014):

- *Complexity*. In the process of "smartening" the grid, it is necessary to deploy some technologies in order to assure the communication among all the grid members and to make the SG possible. However, in the process to increase the degree of communication, the grid operation is complicated as more elements are incorporated in its infrastructure. The main reason of this grade of complexity is the huge amount of data that grid operators manage for a precise functioning of the system. In addition, the integration of new agents such as renewable energies, distributed generation or storage systems, further complicates the power system structure and requires their coordination and efficiently linking with the grid.

- *Technology awareness.* Additional infrastructure will be required for the development and operation of SG technologies. The automated control and sensing systems will require a modern communication infrastructure to perform its daily operation. Other elements are also required such as sensors, intelligent electronic devices, distributed energy resources, cyber security devices, advanced metering systems and other end-use devices which need to be added in the present electricity system. Experts have identified the bi-directional flow of communication as the back bone of the SG. However, there are some concerns about adding all these elements. One of them is the additional costs that suppose providing to grid with these elements, since the industry wants to work with traditional methods for safe and guaranteed return on investment. Industry is also worried with the concern that users do not require all the services that the SG technologies offer to them. So, the industry attitude towards the innovation is not enthusiastic and they have fear of adopting newer technologies. Another concern is the maturity of the technologies to be deployed. New technologies are still emerging and they are not yet validated which is slowing down the development of the SG. Another concern with SG technologies inside is the lack of clear standards to support system interoperability. Many proprietary standards are in use today, which are making difficult the development of these technologies. Thus, it is necessary to replace them with open standards to encourage, complete, stabilize and normalize them.

- *Economic.* The deployment of the SG will require huge amounts of investments related to transfer the necessary technology, adequate the infrastructure and communication systems, hire skilled professionals, R&D work and integration of new sources of generation. In spite of the SG benefits, investors are not interested to invest in its deployment until universal standards are adopted and the return on investments are guaranteed. In addition, there are not specific policies and regulations for free markets tariffs and it is a strong political commitment required and global cooperation to organize the dynamics of energy market. Thus, private investment is discouraged frequently due to the lack of revenue uncertainty and sufficient market base. One method to activate the private investment is that governments provide some incentives and subsidies to support the cost burden. This cost burden should be shared among investors, governments and/or potential beneficiaries to alleviate some of these difficulties. Nowadays many developed and developing nations affected by the global recession are struggling to pay for renewal of their entire major infrastructure and are facing financial challenges.

- *Security.* The deployment of ICTs introduces vulnerabilities to the system based on potential cyber attacks. ICTs may be vulnerable to worms, viruses, denial-of-service attacks, malware, phishing and user errors that compromise integrity and availability. Moreover, some aspects of the grid regulations may make difficult to ensure the cyber security of SG systems. Meanwhile, utilities are focusing on regulatory compliance instead of comprehensive security and consumers are not being adequately informed about these risks. It is necessary the adoption of models that work to develop solutions for cyber security, while allowing the use of the data collected. Analyzing and implementing SG security may be a challenging task, considering the scale of the potential damages that could be caused by cyber attacks.

- *Data privacy.* The use of ICTs allows collecting and access data of different kind, such as electricity consumption, usage, production, etc. There are some concerns about the risks related to how that data is used, shared, stored and accessed. These concerns need to be addressed appropriately to gain consumer acceptance and trust. Organizations must establish internal privacy policies and supporting procedures to protect the data coming from the different parts and especially the data from users. The privacy policies must assure some special topics, such as how the information should be retained, distributed internally, shared with third parties, secured against breach, etc. In addition, SG services and products should provide a privacy notification that describes how the data is collected and how that information will be processed. The data access has to be restricted and controlled in order to ensure that only specific individuals can gain access to confidential information. Some technologies can also be useful to enhance the data privacy such as encryption, steganography or aggregation methodologies to remove personally identifiable information from collections of energy usage data.

- *Public awareness.* At early stages of the SG implementation, the educational aspects are vital to the adoption of the SG by the different actors. General public needs to be educated through proper managed awareness programs about the benefits of SG and technicalities of its usage, in order to make them participant through the use of advanced metering and two-way communication available. If this situation is not handled appropriately, it can be detrimental to the development of a market which helps the return of investment of private initiatives. Nowadays, customers are not massively adopting new technologies to manage their electricity consumption due to the lack of understanding and proper incentives. Thus, advocates need to be able to explain and clearly identify the benefits of each SG component to the customers that are the potential key to service success.

- *Policies and regulatory framework.* The traditional regulatory system needs to evolve and be modified, in order to encourage the utilities' investment towards the SG. Current regulations were created long ago and they were appropriate for those times, but now they are obsolete. Traditional regulatory systems are not harmonized and tend to discourage investment in innovative technologies. The key for good policies and regulations is to find the right balance in sharing costs, benefits and risks. The responsibility for achieving this balance lies with regulators and, in some cases, legislators, but must include input from all stakeholders. That is why a collaboration among stakeholders is necessary to impulse the development of a regulatory framework appropriate to the development of the SG. Moreover, specific policies in the consumer side are required to guarantee that they are protected and assure their benefits. The reason is that consumers are typically not provided with either the service options or pricing information needed to manage their consumption. Thus, it is required that the SG customer policies fall into consumer feedback, protection and pricing.

These are the main barriers that are hindering the development of the SG. In order to get the SG deployment and benefit from it, it is necessary to tackle all of them. Consumer awareness, creation of a regulatory framework and economic investments are the keys in the development of the SG, since technologies involved are mature enough. Demonstrators are being developed all around the world to prove the benefits of the SG and validate its possible deployment at large scale, while searching for a cost-effective way to make it feasible. Governments are accelerating the installation of SG pilots with some investment initiatives as an opportunity to renew their grids and as an important commercial opportunity to offer new services. In order to have a better understanding of the current state of pilots around the world, some national initiatives are described as follows:

- *China.* The Chinese government stated in the 12th Five-Year Plan (2011–2015) that the SG development is a national priority for the energy sector (Xu et al., 2014). The drivers for the developing of the SG are the rapid economic growth, the uneven geographical distribution of electricity generation and consumption and the impressive increase of total energy use, which is still dominated by fossil-fuel-based thermal power generation (Ekanayake et al., 2012). China's State Grid Corporation outlined plans in 2010 for a pilot SG programme that maps out deployment to 2030. It has developed a long-term stimulus plan investment in SG which will reach at least 96 USD billion by 2020 (IEA, 2011).

- *India*. The Indian government is driving some SG initiatives as part of an emerging energy policy of central and state governmental entities. These initiatives include the increased grid capacity to meet the growing electricity demand, rural electrification, and optimizing electrical usage through load management and improving operational efficiencies. The Indian SG is emphasizing the need of private investment in energy production/supply and unbundling the power sector from total government control as well as a new regulatory framework to develop all the agents involved in the SG. The priorities to modernize the Indian grid consist of reducing aggregate technical and commercial losses, automation to monitor and control the flow of power to/from the loads on an almost real-time basis, improvement of system reliability, and intelligently managed loads, congestion, and power shortfall. The Smart grid Forum contributes to the Indian SG vision on advisory basis, coordinating with the Smart Grid Task Force. The work groups, in which the two organizations are divided, are: advanced transmission, advanced distribution, communications, metering, consumption and load control, policy and regulation, and architecture and design (Samantaray, 2014).

- *USA*. In the USA, the initiatives of the SG began with a public law published on December 2007 (United States, 2007). In this law, the government supports the modernisation of the electricity transmission and distribution networks to maintain a reliable and secure electricity infrastructure. The USA government is currently emphasizing the need of performing studies to identify characteristics of the current grid, identifying key technologies that are needed, and proposing a plan for technology deployment in both the short and long term, considering technological, societal, economical, and legislative aspects. The investment planned by the government consists of 4.5 USD billion to grid modernisation, 3.48 USD billion for quick integration of proven technologies into existing electric grids, 435 USD million for regional SG demonstrations and 185 USD million for energy storage and demonstrations (IEA, 2011).

- *European Union (EU)*. Meanwhile, the EU through the SmartGrids Technology Platform published its vision of the SG strategy for Europe's electricity networks in 2006 (European Commission, 2006). The vision of the European SG consists of integrating all low carbon generation technologies to encourage demand side to play an active part in the supply chain. The key challenges identified by the SmartGrids Technology Platform that impact on the delivery of the EU-mandated targets for the utilisation of renewable energy, efficiency and carbon reductions by 2020 and 2050, are: strengthening the grid, decentralised architectures for system control, delivering communications infrastructure, enabling an active demand side management, integrating intermittent generation, enabling DG and storage systems, and preparing for the EV (Ekanayake et al., 2012). Thus, the different countries inside the EU are developing different parts of the SmartGrids Technology Platform key challenges. For example, in Germany, the E-Energy funding programme has several projects focusing on ICTs for the energy system. In France, the distribution operator ERDF is deploying 300000 smart meters in a pilot project based on an advanced communication protocol named Linky. In Spain, the government mandated distribution companies to replace existing meters with new smart meters in 2008; this must be done at no additional cost to the customers. The replacement of these smart meters has to be done before 2018 for all the Spanish territory. In Italy, the Ministry of Economic Development has also granted over 200 EUR million for SG demonstrators and network modernisation in Southern Italian regions. And also the success of the Telegestore project has awarded eight tariff-based funded projects on active medium voltage distribution systems, to prove advanced network management and automation solutions necessary to integrate DG (IEA, 2011).

- *UK*. The UK Department of Energy and Climate change in the UK Department of Energy & Climate Change (2009) identifies key aspects to the SG development in the UK. Rapid expansion of intermittent renewable and less flexible nuclear generation in conjunction with the retirement of flexible coal generation, electrification of heating and transport, penetration of distributed energy resources which include DG, DSM and storage and increasing penetration of EV (Ekanayake et al., 2012). In addition, the energy regulator OFGEM has an initiative called the Registered Power Zone that will encourage distributors to develop and implement innovative solutions to connect distributed generators to the network. OFGEM has set up a Low Carbon Networks fund that will allow up to 500 GBP millions support to projects that test new technology, operating and commercial arrangements (IEA, 2011).

- *Japan.* The Japanese government declared in 2009 that the CO_2 emissions will be reduced a 75 % by 2020. In order to achieve this target, the Ministry of Economy, Trade and Industry set three committees to look into the SG. These committees centered their efforts in studying low-carbon power system, next-generation transmission and distribution networks and how can be deployed the SG in the Japanese context (Ekanayake et al., 2012). The Federation of Electric Power Companies of Japan is developing a SG that incorporates solar power generation by 2020 with government investment of over 100 USD million (IEA, 2011). The project is called "The Field Test Project on Optimal Control Technologies for the Next-Generation Transmission and Distribution System" and was conducted by 26 electric utilities, manufacturing companies and research laboratories in Japan in order to develop the technologies to solve these problems. The Japanese government has announced a national smart metering initiative and large utilities have announced SG programmes (IEA, 2011).
- *Rest of the world.* The Australian government announced the 100 AUD million "Smart Grid, Smart City" initiative in 2009 to deliver a commercial-scale SG demonstration project. Additional efforts in the area of renewable energy deployments are resulting in further study on SGs. In Brazil, a utility association called APTEL has been working with the Brazilian government on narrowband power line carrier trials with a social and educational focus. Some utility operators are developing some strategies to deploy the SG pilots: Ampla is deploying smart meters, AES Eletropaulo has developed a smart grid business plan using the existing fiber optic backbone and CEMIG has started a SG project based on system architecture developed by the IntelliGrid Consortium. In South Korea, the government has invested in a pilot programme on Jeju Island of 65 USD million, which consists of a fully integrated SG system for 6000 households, wind farms and four distribution lines. South Korea has announced plans to implement SG nationwide by 2030 (IEA, 2011).

To sum up, the SG is a new field of development for old power systems to evolve in a new stage where all the agents can be part of a beneficial environment. SG is still at an early stage to be deployed and a lot of research has to be done before it arises at its final stage. Although there is no clear definition of what should be a SG, there are five key aspects in which all the different definitions converge, Figure 2.16 shows them. The SG is based in the actual *grid* that integrates *renewable energies*, *storage systems* and *demand side management* thanks to the *information and communication technologies*. It is in the convergence of these five aspects that the innovation arises and their confluence makes the members of the entire system to benefit from it, as well as the environment. However, it also faces a lot of barriers, although the most worrying problem is the lack of investment to carry out the development of these technologies. Currently, initiatives by organizations and government parties are supporting young drivers and improvements in existing power grid as part of its renovation to approach to the SG.

For a better understanding, each key aspect of the SG is described. The grid was actually defined in Section 2.1, the communication technologies are presented in Section 2.3.1, then in Section 2.3.2 the DG technologies are explained, after that the use of storage systems is presented in Section 2.3.3 and finally in Section 2.3.4, the user importance in the demand curve is described.

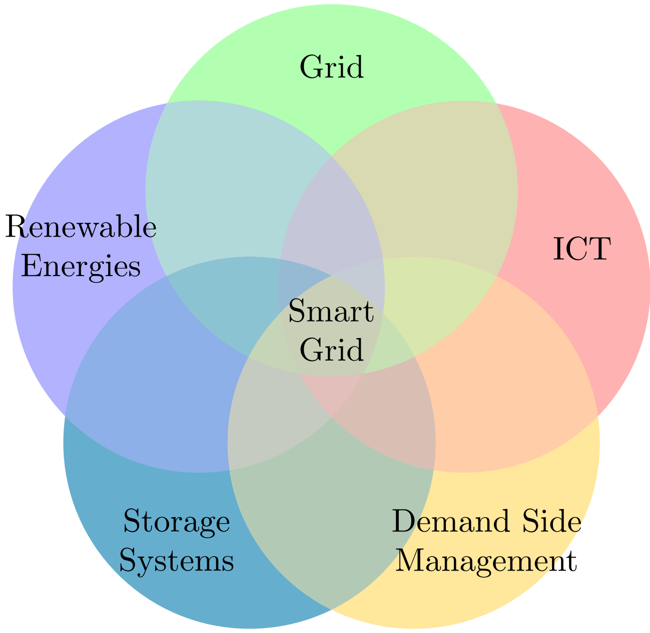


Figure 2.16: Key aspects of the SG

2.3.1 Information and Communication Technologies

Currently, the communication infrastructure of a power system typically consists of a Supervisory Control And Data Acquisition (SCADA) with dedicated communication channels among system control centres and substations, and a Wide Area Network (WAN) for the rest of communications. The SCADA system is dedicated to connect the system operation facilities, such as central generation stations, transmission and distribution substations to control centres. An example of the data that can be sent is the status information from different devices of large facilities to a workstation. While the WAN is used for sending market information and doing business operations. Any operation of the power system relies on effective communications and involves a large amount of connections among the devices and control centres (Ekanayake et al., 2012; Panajotovic et al., 2011).

With the arrival of the SG, the Information and Communications Technology (ICT) is outlined as the technology in which current power systems will be based to evolve its existing communication platforms. The role of the ICT sector in SG has been summarised in a report issued in European Commission (2009). The ICT is a set of technologies that will store, retrieve, manipulate, transmit or receive information electronically in a digital form. ICT will play a major role in the future SG to manage and control the power grid. They will be part of the future infrastructure to provide the grid with a two way communication, in which the flow of information is managed and gathered from each part of the grid to those places where it is required taking actions to enhance the operation of the power system. This will provide the distribution network with a communication infrastructure to manage the data from different users, allowing the ones that produce electricity locally to become "*prosumers*" (consumer and producer of electricity).

The deployment of ICT will consist of unifying the current infrastructure of communications of the grid (SCADA and WAN) and extend it to the rest of its entities. Thus, the SG ICT platform can be divided in three parts depending on the area that they cover: i) WAN, covers a wide area and interconnects substations and control centres, ii) Neighborhood Area Network (NAN), in charge of covering the areas served by the distribution network and iii) Home Access Network (HAN), covers the last part of the system and the communication with the user. The interface between

Sub-network	Communication technologies
HAN	Ethernet, Wireless Ethernet, Power Line Carrier (PLC), Broadband over Power Line (BPL), ZigBee
NAN	PLC, BPL, Metro Ethernet, DSL, EDGE, HSPA, UMTS, LTE, WiMax, Frame Relay
WAN	Multi Protocol Label Switching, WiMax, LTE, Frame Relay

Table 2.2: Technologies used in different communication sub-networks.

NAN and HAN will be done through smart metering (Ekanayake et al., 2012). The ICT platform can be also divided in two parts (Fang et al., 2012):

- *Logical*, includes the data itself and its called the information infrastructure. From the information perspective, the ICT will provide support for sensing and monitoring the different grid entities and manage those informations to model, analyse, integrate and optimize the system. The information has also to guarantee a Quality-of-Service (QoS), related with reliability, latency and network throughput, remote maintenance and configuration, and security requirements. Information will be gathered through sensors such as smart meters to obtain the information from the end users. Smart meters will allow automatically collecting diagnostic, consumption and status data and transferring that data to a central database for billing, troubleshooting, and analyzing them. Thus, all the information is available in real-time and on demand, enabling any system operation or giving users enough information to modify their consumption, prioritizing it based on a target such as electricity prices or grid efficiency. The information collected from the different sensors spread all over the power system will help to asses the real-time mechanical and electrical conditions of the system, obtaining a complete physical picture, diagnose faults, and determine appropriate control measures that could be automatically taken and/or suggested to the system operators (Fang et al., 2012).

  However, huge amounts of data will be generated with the installation and expansion of a massive communication platform. These data contain information from the different entities on the grid, but the data themselves are useless. It is required information management to make data models, information analysis, integration, and optimization of the system from collected data. Using algorithms will help to extract useful information from data and create models to the interoperability of the system (Ribeiro et al., 2014). In addition, increasing the computing capacity of the SG will facilitate the data analysis to actuate based on the data provided from the sensors.

- *Physical*, consists of the connectivity and how the data are transmitted, it is also called communication infrastructure. The implementation of this two-way communication infrastructure is not yet developed. There are several technologies that can be used to develop this communication infrastructure. The lack of system standardization is favoring that each part of the communication infrastructure could adopt freely any technology available. An example of the different technologies that each sub-network implements are listed in Table 2.2. The communication infrastructure uses different wired and wireless technologies (Ekanayake et al., 2012).

  Among the wired technologies, PLC and BPL are communications in which the data is transmitted simultaneously with the electricity inside the conductors. PLC is conceived for LV and communication with users while BPL is used in LV and MV to connect different devices of the distribution network. Another important wired technology used is fiber optic, which is being considered a suitable option thanks to its deployment as communication network for the internet. Among the wireless options are worthy to distinguish the standards IEEE 802 series, which are a family of standards to support Local Area Networks (LAN) (Wireless LAN - IEEE 802.11, bluetooth - IEEE 802.15, WiMAX - IEEE 802.16). In addition, another communication network already deployed that can be used by power systems is the cellular communication systems, consisting of different generations EDGE, HSPA, UMTS, LTE, etc. Other wireless communication technologies are the cognitive radios, microwave communications and satellite communications. There exist lots of standards on communications that could be part of the communication infrastructure of the future power system. This is causing serious concerns about the interoperability of this variety of systems inside the communication platform and how to assure the integrity of the data (Fang et al., 2012).

Coming back to the architectural design of the communication system, it must be able to interconnect the different sub-networks: enterprise bus (connects control centers, markets and generators), WAN (covers the transmission network), NAN (covers the distribution network) and HAN (covers the users). The communication channel, which is the path where data travels as a signal, has to connect the different entities of the communication system. The communications normally are point-to-point but the channel between the source and destination could be dedicated or shared medium depending on the sub-network and the technology used. Communication channels are characterised by their maximum data transfer speed, error rate, delay and communication technology used. Thus, the communications infrastructure, used by power system, could consist of private utility communication networks (radio networks, meter mesh networks) or public carriers and networks (Internet, cellular, cable or telephone) (Panajotovic et al., 2011).

In conclusion, the ICTs are key to stablish the communication necessary to deploy the SG. It is required that the ICTs give coverage to all the members of the system, that is one of the reasons to use private and public communication channels. However, the security and privacy of the channel must be assured, ensuring the protection of the data and not suffering any cyber-attack to cause failures or users could be in danger. Not using a common set of standardized communication is making complex the system structure, that is why the interoperability must be guaranteed inside the ICT infrastructure. Finally, the ICT infrastructure needs to guarantee the QoS because of the existence of critical data (e.g. the grid status information) that must be delivered promptly.

2.3.2 Renewable Energies: the Distributed Generation paradigm

Once that the bidirectionally of the information is assured by ICT infrastructures, it is possible to take advantage of new generation technologies spread out all over the grid. With this new communication system, a door is opened to integrate all the generation technologies inside the SG. So far, only huge isolated power plants were used as the main power source of electricity. However, with the rise of different renewable energy technologies in power systems, this paradigm is rapidly changing due to the possibilities that they offer. Renewable energies are understood as the set of energies that come continuously from resources which are naturally replenished on a human timescale and are derived from the sun, directly (thermal, photo-chemical, and photo-electric) or indirectly (wind, hydro and biomass), or from natural movements and environmental mechanisms (geothermal heat and tidal). Renewable energies do not include energy resources derived from fossil and nuclear fuels (Ellabban et al., 2014).

According to the European Renewable Energy Council, the renewable energies are theoretically able to provide about 3000 times the total current world energy consumption (Zervos et al., 2010). Figure 2.17 shows how these flows of energy are shared by different possible sources. The solar resource represents more than 90 % of the total. The use of renewable energies has been extended in the last decades to

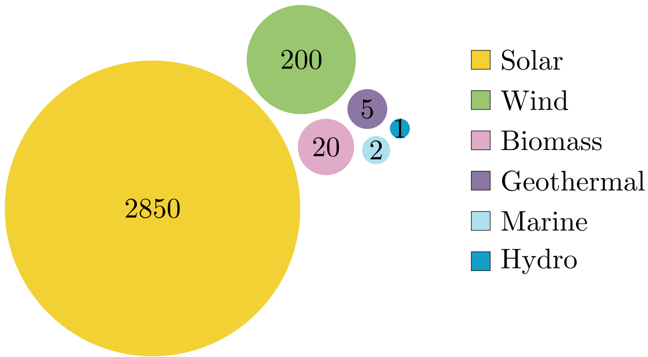


Figure 2.17: Renewable energy resources in the world in terms of the available resource.

replace the traditional uses of fossil fuels mainly in the power generation sector, but most recently also in the heat production and transport. One of the reasons of its increasing use is that the natural resource is almost infinite. This favours consuming less fossil fuels, avoiding energy-related CO_2 emissions. Moreover, being cleaner and safer for the environment are the other two key factors of its increasing use (IEA, 2014d).

Among the different technologies that take advantage of these resources, the one long used is the hydro power, which uses hydroelectric turbines to convert the downhill movement of water into rotation of an electric generator. Tidal power is similar to hydro power but with the difference that the turbine are submerged in a tidal basin instead of a river. Then, there are some renewable technologies that use steam turbines to produce electricity like solar thermal (uses the sun to heat a fluid), biomass (uses organic material that burns) and geothermal power (where steam or hot water is extracted directly from the ground). On the other hand, there exist another electrical generation technologies that differ from the traditional ways of generation: wind turbines, Photovoltaics (PV), microturbines and fuel cells. They also differ from the previous one in the electrical properties of the generator component, the availability of the resource, the customizable scale and the range of suitable locations for their deployment. Wind power uses the wind to move turbines, which are based on induction generators. The rotor varies its velocity depending on the wind speed, it also uses a condition power stage to control the speed and the electrical parameters of the electricity produced. Microturbines are powered by natural gas coming from organic sources (wastewater, landfill or manure digester gas) and are similar to steam turbines but smaller. They are usually used as a cogeneration system in which the heat and the power are exploited. Finally, PV and fuel cells produce electricity in a totally different way than the previous ones. They have no rotating parts and produce DC rather than AC electricity, so they use an inverter to connect them to the grid. PV uses a treated semiconductor to produce an electric potential when it is exposed to the light. Whereas, a fuel cell converts the chemical energy from a fuel into electricity through a chemical reaction of positively charged hydrogen ions with oxygen or another oxidizing agent. The output voltage of a PV and/or a fuel cell depends on the characteristics of the material, but normally it is on the order of 1 V. Therefore, various cells are usually grouped together to achieve a reasonable operating voltage (von Meier, 2006).

New renewable energies allow configuring different sizes of generators due to the modularity of its components (PV or fuel cells) or its miniaturization (smaller wind and hydro turbines). This fact is making possible the rise of new generation forms different from classical large power plants far from consumption places. Users are investing to install some small generators in their houses and businesses to supply their own electricity. Thus, the generation paradigm is evolving to smaller generators in scale, geographically distributed across the grid and closer to the places where it is consumed (Pepermans et al., 2005). These new forms of generation are what is known as Distributed Generation (DG).

The decentralization of the generation have great benefits for the grid. The system support benefits are as follows (Pepermans et al., 2005; Barker and de Mello, 2000; von Meier, 2006):

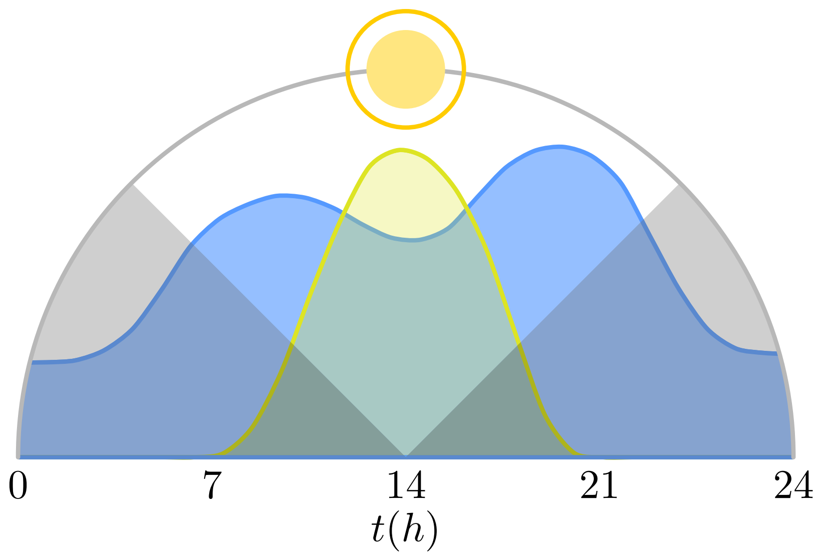


Figure 2.18: Example of PV generation with aggregated consumption of a power system. In blue the aggregated consumption of the power system, in yellow the DG and in shadow grey the night hours are represented.

- *Loss reduction.* By generating electricity closer to the places where it is consumed, the electricity does not need to be transported large distances, so the thermal losses of the power lines decrease.

- *Reliability and power quality.* Generating electricity in more than one place adds redundancy in the system that helps to enhance its security. For example, it helps to overcome easily sustained interruptions (voltage drops to zero) or sabotages along the power system. Related with power quality[3], DG can increase it by providing voltage support to rise the insufficient voltage of an area or providing reactive power support to compensate it. So, if DG can provide voltage support reliably, installation or upgrades of new hardware can be avoided.

- *Avoiding upgrades in the transmission and distribution network.* The installation of DG reduces the constraints of the transmission and distribution networks, alleviating the power system size. The equipment inside the network has to accommodate the peak load, but offsetting a demand amount with DG can provide relief in the infrastructure (e.g. for a PV DG can be observed in Figure 2.18). Thus, it is not necessary to invest in order to upgrade the infrastructure while maintaining reliability.

- *Grid support.* DG could also contribute to provide auxiliary services to maintain a stable grid operation, such as generation on demand by operators, stabilising the frequency, etc.

[3]Power quality is referred to the waveform of i and v compared to an ideal sinusoidal one and the balance between them.

- *Economic savings.* These new technologies offer some economic reductions. The investment in grid infrastructure can be reduced, as well as reducing oversizing related to system demand or investment in new equipment to ensure system reliability. Also, users can benefit from using DG locally by reducing the purchase of external resources. In addition, DG can be used depending on the electricity pricing, following an economic saving strategy. The costs are also shared by the different grid stakeholders, resulting in a collective benefit for all of them.

- *Environmental friendly.* Apart from being increasingly cost-effective technologies, they also have some environmental benefits. In general, DG consists of renewable energies which are low emissions power supplies. Thus, the dependency from environmentally damaging fuels is reduced. Consequently, the levels of greenhouse gas, toxic chemical emissions and other damaging emissions are also reduced. For example, cogeneration technologies use waste materials so the need of landfill sites are reduced.

On the other hand, DG faces important challenges for its fully adoption. The first one is the high investment costs of some technologies. The problem is that for some technologies there are no economies of scale, which makes difficult their deployment. Moreover, there is not a clear financial credit to support the installation of DG in spite of being considered as key technologies to decrease the harmful effects of electricity generation to the environment. Another issue is the need of extra protection and coordination of the generation resources. The reason is that the flow of electricity is now bidirectional, consequently the system is no longer a centralized, radial and flat system. At the distribution level there is not only consumption, but also generation is injected at this level. Thus, the equipments inside the distribution network can experiment a flow of energy in the other direction for which they are not built. It is necessary to introduce extra security elements such as switches or even change the equipment to support the bidirectionally flows of energy. The coordination issue is related with control and availability of resources. The problem of some renewable energies is that they do not produce regularly the same amount of electricity and depend on the weather and other variables which are difficult to predict. This fact makes difficult the generation scheduling and increases the operation complexity of the grid (Pepermans et al., 2005; von Meier, 2006). Finally, there are also some environmental concerns related to certain renewable energies. For example, some organic fuels of biomass are crops destined to produce energy and some markets are changing the way of cultivating some areas. Other concerns are related to the visual impact of the technologies in the environment, such as large wind turbines or huge PV solar fields (Ellabban et al., 2014).

In spite of these deterrents, all these problems are minor drawbacks overcome by the benefits that the DG possesses. The problem of the generation scheduling is not crucial since it can be tackled in various ways, using predictive algorithms for proper system integration or trying to design a system to meet the demand without any surplus. About environmental, financial costs and protection issues, regulations and policies are the elements responsible for changing the actual situation and favor the deployment of DG.

The election of a renewable technology in DG is decided based on how near it is to the loads, however, it must also be environmentally compatible. Among all the renewable energies available, the PV is the more suitable to almost any application. PV is easy to deploy because it involves the least possible inconvenience: zero emissions, no noise, reduced aesthetic impact, different options can be used to integrate it in buildings or other structures (for example, sunshade for windows or shading for parking lots), easy configuration of the generator adding or eliminating PV panels and low maintenance of facilities (von Meier, 2006). Other technologies such as fuel cells require more supervision, or wind power requires better conditions for resource availability.

To sum up, the DG of renewable energy is presented as one of the alternatives to our present centralized and hierarchical power system. Many questions remain to be solved regarding their availability and their development to achieve greater efficiencies, as well as better policies that support its use. The use of these technologies have societal implications beyond the electric power as they are related to resource scarcity, environment, and the politics of ownership (von Meier, 2006). In order to address some challenges related to the control of these technologies, it is necessary to combine them with the use of storage systems, to alleviate the availability problem, which is the subject of the next Section.

Type	Technology
Mechanical	Pumped hydro, compressed air and flywheel
Electrochemical	Secondary batteries (Lead acid, NiCd, NiMh, Li and NaS) and flow batteries (redox flow and hybrid flow)
Chemical	Hydrogen: electrolyser, fuel cell and SNG
Electrical	Double-layer capacitors and superconducting magnetic coil
Thermal	Sensible heat storage

Table 2.3: Classification of EES systems according to the energy form of storage.

2.3.3 Storage Systems

Electrical Energy Storage (EES) is expected to play different and important roles in the future SG. In the current power systems, the electricity is consumed at the same time that it is generated. However, adding EES will relax the operation constraints, being not necessary that generation meets consumption in real time. The use of storage system is a great asset to integrate inside the grid, because it can be deployed in any part of the power system, helping with the integration of DG and guaranteeing the supply during peak demand periods or when resources are unavailable, working as a backup system. An EES can store energy in a wide power range, supply it in a wide range and tolerate repeated discharges, depending on the technology. Despite this, the use of storage in grid-connected applications is not very common (IEA, 2014a).

The EES presents interesting features for power systems, such as modularity, controllability and responsiveness. Storage systems of different sizes can be built adding more or fewer elements easily depending on the technology. It is also easy to retrieve the energy stored in the system to supply a load within a very short time (IEA, 2014a). Due to these properties, EES can be integrated in any part of the grid easily. However, depending on the application, some EES technologies are more suitable than others for the task development. There exist multiple technologies to implement an EES system. A possible classification of the different technologies is shown in Table 2.3 (IEC, 2011). This classification is done attending to how the system stores electricity. Since storing directly electricity is a difficult process, it can be easily done in other forms and converted back to electricity when needed.

- *Mechanical storage systems* consist of using the potential or kinetic energy of a body to store electricity. Pumped hydro and compressed air use the potential energy of the water or air, respectively, to move a turbine and produced electricity. These systems are largely used and represent the 99 % of all deployed electricity storage (IEA, 2014a), which is around the 3 % of the global generation capacity (IEC, 2011). Compressed air systems are similar to pumped hydro systems, but they pressurize air in a reservoir during off peak hours and release it when needed. On the other hand, a flywheel stores energy through accelerating a rotor and maintaining it as inertial energy. The energy is maintained by keeping the rotating body at a constant speed and it is released reversing the process and using the motor as a generator (Mohd et al., 2008). Mechanical storage systems can store electricity up to the order of MW, but they are large systems and require considerable maintenance.

- *Electrochemical* type of EES are also known as batteries and they are along with pumped hydro systems the oldest forms of EES. The electricity is stored chemically inside the battery. Secondary batteries consist of a liquid, paste, or solid electrolyte together with a positive electrode (anode) and a negative electrode (cathode) (Mohd et al., 2008). During the discharge process, electrochemical reactions occur in the two electrodes and a current is generated between them. Reversing the process will charge the battery. There are different chemicals used to build the battery: lead acid are the oldest and most widely used, nickel cadium (NiCd) is also quite widespread, sodium sulphur (NaS) and lithium ion (Li-ion) (Chen et al., 2009). Li-ion are becoming an alternative to the oldest batteries due to its efficiency (around the 100 %), the increase life cycle of the battery and its reduced shape for some applications (Chen et al., 2009). Flow batteries are also rechargeable batteries, but the energy is stored in one or more electroactive components, which are dissolved in liquid electrolytes. The electrolytes are stored in tanks and pumped through the electrochemical cell that converts chemical energy directly to electricity and vice versa (IEC, 2011). There are two main types of flow batteries: redox and hybrid flow batteries.

Mechanical and electrochemical storage systems are the oldest types of EES and are already mature enough to be installed. However, new types of EES are in development, such as the hydrogen systems. These storage systems consist of storing the hydrogen generated in an electrolysis process. Once the hydrogen is produced, different alternatives are available for its use as an energy carrier: transport, heating, chemical industry, etc. The problem is that the efficiency is low (around 40 %) compared to other technologies, but it can store large amounts of energy (IEC, 2011). Other EES systems are thermal ones, in which the electricity is used to heat or cold a fluid and later used in different applications, such as space heating or cooling, hot water production or electricity generation (Mohd et al., 2008). The last type of EES stores electricity directly: super capacitors and superconducting magnetic coils. The first one has the features of batteries and capacitors and electricity is stored in the form of an electric field between two electrodes (Mohd et al., 2008). They are very durable (8–10 years) with high efficiency (∼95 %). However, they present a high self-discharge so energy has to be used quickly (Mohd et al., 2008). On the other hand, the superconducting magnetic energy storage consists of storing energy in a magnetic field created by the current flowing through a superconducting coil (IEC, 2011). They possess a fast response, but they require high operating costs, deep discharges and constant activities (Mohd et al., 2008).

Some features to consider the election of a storage system are its maturity, life cycle, costs, efficiency, environmental implications and density. Depending on the technology some of these features are better than others. For example, pumped hydro, fuel batteries or fuel cells are suitable for energy management application, whereas flywheels, batteries and supercapacitors are more suitable for power quality and Uninterruptible Power Supply (UPS) applications. EES provides the grid with some advantages, such as (Mohd et al., 2008; IEC, 2011; Chen et al., 2009; Wade et al., 2010):

- *Voltage control.* Maintaining voltages within acceptable range implies that it provides load factor correction, reduce the need to constrain generation, mitigate flicker, sags and swells, etc.

- *Frequency support.* Reducing the imbalances between generation and consumption favors keeping the grid frequency within its limits for periods up to 30 minutes.

- *Transient stability.* Oscillations are reduced in the power signal by injecting or absorbing real power.

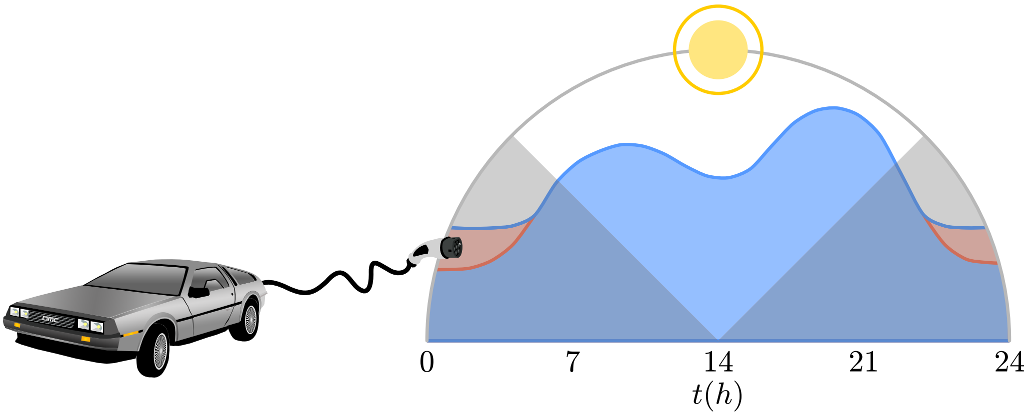


Figure 2.19: EV charging for load levelling during valley hours. In blue the aggregated consumption of the power system, in red the valley filling technique using the EV and in shadow grey the night hours are represented.

- *Power quality and reliability.* Power quality is related with changes in magnitude and shape of the voltage and current. Storage systems can help with harmonics, power factor, transients, flicker, etc, providing reliable electricity to consumers. EES could supply the consumption during interruption or blackouts, but also it could help to restore the network after faults.
- *Load levelling.* A storage system is charged with excess power during low-demand periods and it is discharged when the power demand is on peak levels.
- *Spinning reserve.* The power of an EES system is reserved to supply the load immediately when a fault occurs.
- *Power flow management.* Redirecting power flows increases operation efficiency and cost reductions in fuel. This property also contributes to the delay network reinforcement, reduction of reverse power flows and minimization of losses.

In addition, EES could help in the energy market by arbitraging and balancing it. It could also help to reduce the variability of the DG, increase its yield from non-firm connections, compliance with energy security standards and support network management (assist islanded networks, support black starts, etc.) (Wade et al., 2010). The main reasons of storage absence in the grid are typically its costs and the generation control possibilities of present power systems. It makes unprofitable the deployment of new storage systems on a large scale. However, high investments in EES can be affordable when these costs are offset by higher revenues, such as purchasing inexpensive electricity during low demand periods and later selling it to a higher price. Moreover, it could reduce financial losses associated with power quality anomalies and outages, increase the revenue from renewable energy sources, reduce the variability of the generation, increase revenue for ancillary services, and avoid installing extra generation (Mohd et al., 2008). There are also some concerns about the maturity of the technologies, its energy density, power capacity, life cycle, efficiency, discharge capability and environmental issues. Only two technologies are currently mature enough, pumped hydro and lead-acid batteries, the rest are currently under development, although some technologies are almost mature such as Li-ion batteries (Chen et al., 2009). With respect to the efficiency, EES have high efficiencies depending on the technology, batteries, pumped hydro, compressed air and supercapacitors present an efficiency above 60 % (Chen et al., 2009). Although the development of EES is advancing, the life cycle and the discharging capacity are still not enough for its integration in some parts of the grid (IEA, 2014a). EES presents some negative effects on the environment, fossil combustion, strong magnetic field, landscape damage and toxic remains (Chen et al., 2009). For example, the remains of the batteries are toxic and must be treated in order not to damage the environment. But it is still necessary a deeper analysis of the environmental implications of installing the storage system, taking into account the installation and use of the system. Despite all drawbacks found, EES is needed in the grid due to the flexibility that provides to the power systems. That is the reason why the storage level is expected to grow to 10–15 % from the current 3 % in USA and European countries, and even higher in Japan in the near future (Chen et al., 2009).

Among EES systems, secondary batteries are gaining special importance because of two factors: integration with DG powered by renewable energies and the Electric Vehicle (EV). The use of batteries with DG will improve the usability of renewable energy, since it will help to reduce the variability of the resource, charging the battery with the electrical surplus and making it available when it is needed. On the other hand, the EV market is growing and it is expected to continue this trend in the next decades (IEA, 2013). The reasons behind this growth are mainly due to lower battery costs, reduction of CO_2 emissions and reduction of the dependency from fossil fuels (IEC, 2011; IEA, 2013). The electrification of the transportation sector is associated with a massive deployment of EV in the coming years. This electrification supposes a paradigm shift in energy systems because the grid must be able to absorb a major and new electric consumption. The penetration of the EV will have a great impact on the distribution networks. And if it is not handled with care, it can lead to negative effects (II et al., 2011). The EV charge could become a serious problem due to the increase of power from the distribution level and could affect transmission network or even to the generation capacity of the grid causing an overall increase of the required infrastructure. However, the EV charge offers a flexible load that the grid could use to enhance its functionality. Thus, these problems can be addressed from EV charge coordination (Deilami et al., 2011; Mets et al., 2010). For example, in Figure 2.19, the charge of the EV is used to reduce the variability of the consumption, charging it at off-peak periods. Managing the charge of EV could support the grid and improve its operation through the reduction of power losses levelling the load (Deilami et al., 2011).

Finally, the integration of EES and in particular the EVs in the grid is increasingly feasible thanks to its participation in the SG. Due to the ICT platform, the charge of the EV can be easily done by the grid. With Vehicle-to-Grid (V2G), EVs will be not only mere consumers but they will be used to provide power for specific electric markets. The EV charging management can be framed in the general concept of the DSM, which is discussed in Section 2.3.4.

2.3.4 Demand side management and electricity management strategies

So far, new elements have been presented for their integration within power systems to enhance them. However, the potential of these elements alone could benefit or harm the grid operation, depending on how they are managed. Thus, it is necessary to provide grid operators with a number of tools to coordinate the new power flows that appear from the integration of DG renewable energies and EES together with ICTs to reach them. It is in this framework in which users will play an important role in the SG, by helping to manage their demand towards energy efficiency.

Therefore, an increasing number of programs and policies are motivating users to collaborate with the grid operators for its enhancement. All these programs are included within the concept of Demand Side Management (DSM). This concept is not new, it appears in the 80s and was defined as "planning, implementation, and monitoring of those utility activities designed to influence customer use of electricity in ways that will produce desired changes in the utility's load shape, i.e., changes in the time pattern and magnitude of a utility's load" by Gellings (1985). Currently, DSM is defined as actions that influence the way that consumers use electricity in order to achieve savings or higher efficiency in energy use. DSM is a global term that includes a variety of activities, such load management, energy efficiency, etc. in a long-term perspective. Whereas the short-term perspectives are tackled by Demand Response (DR) programs, which are actions that result in short-term reductions in energy demand (Albadi and El-Saadany, 2007). DR can also be defined as incentive payment programs to motivate users to respond to changes in price or availability of electricity by changing its consumption (Gelazanskas and Gamage, 2014). The different DSM techniques are focused on the modification of the load shape. In general, these techniques seek the grid enhancement through an increase of the energy efficiency. Six different load shaping categories can be distinguished (Gellings, 1985; Gelazanskas and Gamage, 2014) (see Figure 2.20):

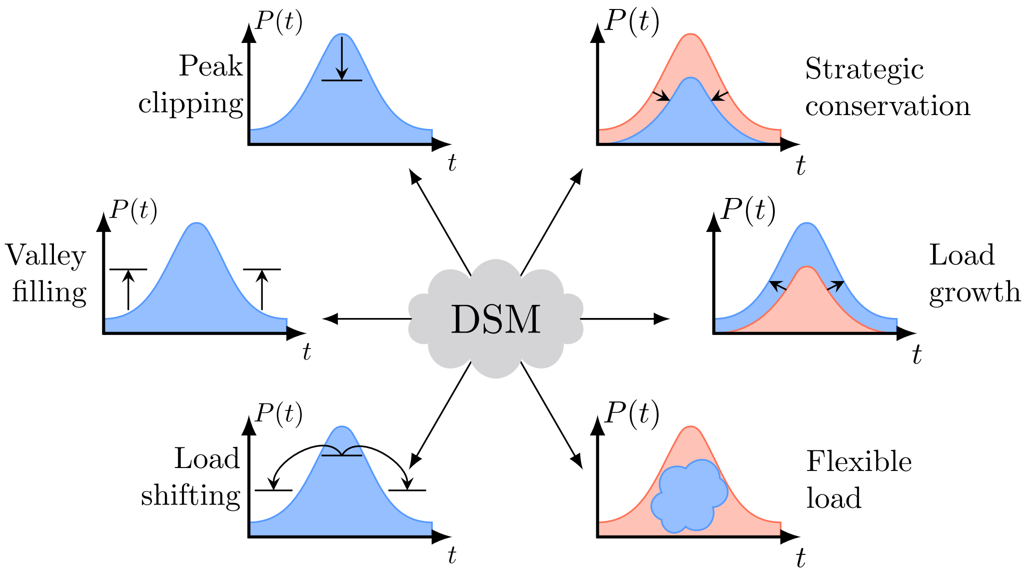


Figure 2.20: Load shaping techniques of the DSM.

- *Peak clipping.* The consumption peak is one of the major problems in the design of the grid capacity and one of the reasons of the oversizing problem. This peak can be reduced through DSM by using direct load control. The supply interruption of some consumers by grid operators and the load automation are the main techniques to implement this effect in the aggregated consumption.

- *Valley filling.* This technique considers the increase of the consumption during the off-peak periods or valleys to reduce the variability of the aggregated consumption (see Figure 2.20). Increasing the consumption in the valley can be accomplished with the connection of pumping stations, EES or EVs during the low consumption periods, among other measures. Valley filling implies an improvement on the grid profitability because of a greater use of its infrastructure.

- *Load shifting.* This consists of displacing part of the electric loads from peak to valley periods. It may be implemented through EESs, user load shift or through a variable pricing strategy depending on the aggregated consumption shape or the electric load automation. The load shifting combines the benefits of the valley consumption increase and the peak consumption decrease.

- *Strategic conservation.* It consists of the consumption reduction without modifying the shape of the aggregated consumption. This effect can be produced through the efficiency improvement of the consumption appliances or increasing the society energy awareness. The consumption reduction leads to a reduction of the grid size but does not enhance its efficiency.

- *Load growth.* In this case, the consumption is increased motivated by the electricity market or other incentives by the utility. Load growth is motivated by the electrification and absorption of new loads that previously did not use electricity, such as EV. It is related with the development of some sectors (transportation or industrial) to reduce the use of fossil fuels and raw materials resulting in improved overall productivity.

- *Flexible loads.* It is related to planning the shape of the load to respond to a particular objective. Users can make flexible their loads through DSM if they are presented with options, such as pricing or variations of QoS, in exchange for various incentives. Load control will offer individual customers the possibility to make their load flexible enough to guarantee incentives.

From the DSM perspective, it is interesting to divide the consumption in terms of its controllability. Thus, the consumption is divided in: i) fixed, ii) deferrable and iii) elastic (Castillo-Cagigal et al., 2011b). *Fixed* or non-deferrable loads represent the consumption that is uncontrollable. They are consumptions that respond instantaneously to user requirements and their use is immediate (e.g. television, computers, lighting, etc.). *Deferrable* loads are consumptions that can be displaced in time but the amount of energy that they consume is fixed during a certain time period (e.g. dishwashers, washing machines or dryers) (Ramchurn et al., 2011). *Elastic* consumption represents loads whose instantaneous power can be controlled directly (e.g. air conditioning or EV) (Ramchurn et al., 2011).

The increase of load automation is required to apply DSM techniques and obtain higher benefits, enhancing the grid operation (IEA, 2011). Some benefits that can be achieved through the use of DSM techniques, are as follows (Strbac, 2008; Palensky and Dietrich, 2011; IEA, 2014d):

- *Reduce infrastructure oversizing.* The capacity of the grid is directly proportional to the maximum demand of the consumers. By reducing the consumption during peak periods, the generation and transmission and distribution lines may reduce their oversized capacity. DSM provides a virtual storage system with fewer costs and the right incentives. Economically, the installation and maintenance costs are reduced by using them.

- *Improve grid investment profitability.* Shifting the consumption from peak periods to off-peak ones, reduces the variability of the consumption, enhancing its operation and favoring for a greater use of the electricity infrastructure. It implies a faster return of investment if it is translated into economic terms.

- *Security and ancillary services enhancement.* The deployment of DSM techniques leads to the development of ancillary services, such as increase monitoring platform, control capability of the grid, distributed spinning reserves, load as virtual storage power plants, communication protocols, etc. The increase of the control capacity allows carrying out new emergency actions and services in the grid.

- *Integration of new generation technologies.* The emergence of new technologies, such as DG or EV, inside the grid, brings new management challenges that can be tackled with the help of DSM techniques. The variability of some renewable energies makes difficult their integration in the daily grid operation. Thus, it is necessary to search for new solutions, such as adapting the demand to the generation through DSM techniques (Castillo-Cagigal et al., 2011b).

- *Integration of new local infrastructure.* The classical structure of the grids consists of a hierarchical structure where energy is produced in large power plants far from consumers. The evolution of this structure to a decentralised one is becoming real with the appearance of the Distributed Energy Resources (DER). DER consist of a wide range of local generators and storage systems which are geographically dispersed, generally close to the consumption, and locally managed with DSM techniques (Castillo-Cagigal et al., 2011a). It is in this framework in which users become "prosumers" who not only consume electricity from the grid but also generate it.

DSM techniques present lots of benefits. Although some real experiments have already been developed (Castillo-Cagigal et al., 2011a; Palensky and Dietrich, 2011), there exist some barriers that difficult and slow their development. The barriers faced by DSM are of different nature. One of these barriers is the lack of knowledge and potential of DSM by users (Strbac, 2008). The hierarchical structure of the grid has lead to a complete ignorance of the system operation from the consumer side and has resulted in a lack of incentives and policies about DSM. There is also a lack of knowledge about the costs and benefits of DSM because of a lack of methodologies to evaluate it. In Torriti et al. (2010) the authors analyze DSM experiences in Europe and conclude that, while it is clear what DSM initiatives can achieve in terms of demand shifting from peak periods, limited knowledge has been developed about its overall energy saving capacities. Because of the lack of measures to analyze DSM effects, it is also difficult to understand the saving impacts on overall financial consumer expenditures. However, DSM programs should cause a high reduction in the energy bill to be attractive to the consumers since the price of the electricity is growing.

Another barrier is the lack of infrastructure. There is no information coming from the grid to the users, only the bill of the electricity consumed in a time period. This fact makes difficult the decision of the users in order to actuate over their consumption. In order to perform DSM, the consumers require having available energy cost information in real-time or other information related to the grid status. Thus, this is one of the reasons to deploy an ICT infrastructure with advanced metering and control methods within the grid. The slow penetration of these technologies in the electricity system and the lack of international and widespread communication standards have slowed the diffusion of DSM. In fact, this has been one of the central barriers to DSM, particularly in the residential sector, where costs tend to be high relative to savings, as compared to commercial customers (Kim and Shcherbakova, 2011).

Another issue related with the DSM is the investment recovery uncertainty. The investment is crucial to the development of DSM. Some investments done by users consist of electronic devices that communicate with the ICT infrastructure and the monitoring network to apply the DSM techniques. However, it is uncertain that the information coming from the grid is available to the users. The recovery of these investments is slow because of the currently relative low electricity prices. Regarding private investments, there are greater uncertainties about the costs and benefits as well as the lack of consumer interest. So, in liberalized energy markets, the lack of revenues is making the private investment lower despite the technological benefits of all the agents (Kim and Shcherbakova, 2011). Therefore, DSM programs must take into account the whole energy industry by increasing their complexity in their development.

The last important barrier in the development of the DSM is the necessity of policies and incentives that motivate its use. The DSM concepts and techniques involve the different grid agents and the necessary development of them implies all the participants of the grid, from big energy producers to small consumers. Thus, it is difficult to develop programs and policies that encompass the entire grid. In addition, the conservative position of the majority of the agents of the electric power sector is slowing down the DSM implementation. But, this situation is changing as governments are taking part in enhancing the grid operation (European Commission, 2006; U.S. Department of Energy, 2009; UK Department of Energy & Climate Change, 2009; Canadian Electricity Association, 2010). And new energy policies are emerging motivated by improving the grid efficiency and fighting climate change (Edenhofer et al., 2011).

The interest of developing these techniques consists of enhancing the grid operation and helping with the integration of new technologies. Figure 2.21 represents the ultimate strategy to achieve with DSM. The grid improvement is done through load shifting and obtaining a flatter consumption, in which all the benefits of DSM can be achieved. Moreover, this smoothed consumption also facilitates the grid operation by knowing exactly at each time the consumption shape and adjusting generation to consumption. This is the objective to achieve at the end of this Thesis through DSM and coordinating the consumption of the different users inside it.

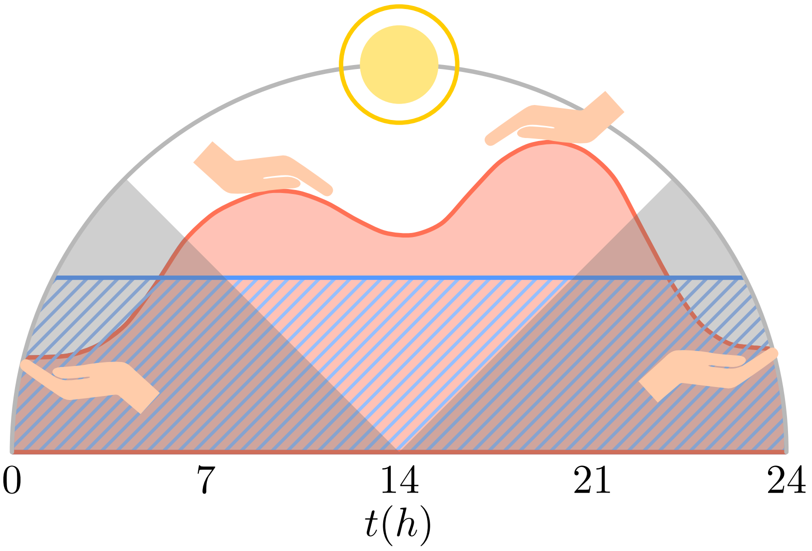


Figure 2.21: DSM to improve the operation of the grid through smoothing the aggregated consumption with load shifting.

At the present, some DSM mechanisms are beginning their implementation, answering users necessities and being carried out with different targets. The objective of DSM mechanisms varies in nature. Some of these mechanisms consists of changing the regulatory framework, encouraging consumer awareness, pursuing a reduction in the electrical appliances consumption, making possible the active participation of consumers in electricity markets, etc. DSM can be classified in the following classes depending on the interaction degree between the consumer and the electric power system (Palensky and Dietrich, 2011):

- *Electricity saving and efficiency programs.* These programs seek to increase energy efficiency of consumption through different DSM strategies. These initiatives provoke an indirect reduction on the long-term demand in terms of power consumption, regardless of the consumption time. Some examples are: incentive campaigns for the use of energy-saving lamps, variable speed drives in electric machines (Mecrow and Jack, 2008), energy policies (Abdelaziz et al., 2011), etc.

- *Indirect control of electric loads through pricing.* In this case, DSM programs seek to lighten the consumption from peak periods through the use of the electricity price signal. The price variations reside in the costs involved for the electrical grid to meet the demand, so peak periods are more expensive than valley ones. And users are encouraged to consume during off-peak periods (Newsham and Bowker, 2010). There exist different time-base rates to incentive the DSM through pricing (Albadi and El-Saadany, 2007): i) Time of Use (TOU) tariffs, the day is divided in different blocks and the price of consuming in each block depends on the historical average costs of electricity during the period covered by the block (e.g. two blocks, one for peak periods and one for off-peak periods), ii) Critical Peak Pricing (CPP) consists of higher prices pre-specified for electricity superimposed on normal rates to periods in which production costs are very high, due to the difficulties of matching supply and demand, iii) Real Time Pricing (RTP) consists of charging customers hourly with fluctuating prices reflecting the real cost of electricity in the whole sale market, and iv) Peak Time Rebates (PTR), these programs consist of discounts on electricity bills for not using power during the peak hours.

- *Direct load control.* These measures consist of directly disconnecting some electrical loads of the users by system operators. Although a few experiences have been reported with residential consumers, the direct control of loads is usually applied in the industry, controlling large consumers such as foundries (Torriti et al., 2010).

- *Smart metering and appliances control.* These programs make users participants to control their consumption through market programs or structures that offer load reductions by adapting the demand to the grid status. To do this, it is necessary that customers have enough information of the grid status and also access to their consumption to modify it. So, some sophisticated customers can already tie the electricity pricing into their energy management system. Moreover, these initiatives can achieve greater benefits and greater load shifts if the price differences between peak and off-peak periods are higher. These techniques require to be accompanied by the application of intelligent appliances that facilitate the implementation of DSM (Strbac, 2008).

Continuing with these last incentive programs, new concepts emerge within DSM framework. Hence, the combination of DSM with an automatic control of demand and local generation leads to a new concept called Active Demand Side Management (ADSM) (Castillo-Cagigal et al., 2011a; Matallanas et al., 2012; Masa-Bote et al., 2014). The use of ADSM strategies could benefit firstly to consumers, provided that the technologies that implement them automate the operation of devices without compromising the users' comfort needs and preferences. In the residential sector, there exist lots of possibilities to implement ADSM, since it can be combined with additional comfort and security functions, improving the demand response, reducing the environmental impacts (Papagiannis et al., 2008) and offering the user information about its electrical consumption. But the residential sector is not the only sector in which DSM strategies can be applied, for example in the service sector, the control of the air conditioning is being explored in order to implement ADSM. Although ADSM does not directly reduce the amount of energy, the feedback of the consumption information to the users could reduce the amount of electricity demanded by them (Wood and Newborough, 2003). At present, ADSM strategies are still under development because it is necessary enough information of local resources (PV or EESs) and the ability to control remotely the different electrical devices to modify the consumption pattern. In addition, some information coming from the grid such as its status or the pricing signal are becoming available through the deployment of the smart metering.

In conclusion, the benefits of DSM represent a major step towards controlling the different loads of the grid and a considerable improvement in efficiency since it can adapt dynamically the consumption to the generation favoring an enhancement of the operation of the grid. Thus, the benefits are not only translated in the electricity bills of the user, but also in the environment through the reduction of emissions and other harmful wastes. In addition, the DSM adds value to the operation of the network since the development of many SG technologies have been deployed and DSM can help control power flows without necessarily increase the complexity of the grid operation. In this Thesis, the use of the ADSM helps to integrate the DG and the use of local storage systems to manage the power flows inside facilities and at the same time to enhance the grid status reducing the variability of the aggregated consumption.

3 Neural Networks

"Computers deserve to be called intelligent if they could deceive humans into believing that they were human" — Alan Turing

These days, there exists a growing concern about the use of the data obtained from different services present in daily live. This is due to the tendency of a society increasingly connected through networks: internet, social (facebook, twitter, etc.), electrical, etc. (Hilbert and López, 2011). In this context, new concepts arise to extend even more the connectivity to all the devices and elements that surround us, such as the Internet of Things (IoT) or the Smart Grid (SG). All these applications produce a vast amount of data which by themselves have no value, but treated properly they can make a difference in the development of society. The information that resides within these data can be made of many kinds: electricity consumption, personal information, interests, mobility along the day, location, etc. Thus, if the information is not treated adequately, it can be a burden for the concrete service or application, causing a loss in quality that can lead to a total collapse in the worst case scenario.

Nevertheless, the access to the information that resides within the data is a major asset to benefit all groups, because it can mean greater operational efficiency, cost reductions or reduced risks. Establishing smart relationships between data and users could enhance the service and offer added value to it, making life more bearable. These new correlations could be the fault preventions in electrical grids, stock prediction in markets, combat crime and so on. Therefore, companies are currently developing tools that facilitate the acquisition, processing and decision-making of data automatically, through what is known as Big Data Analytics (BDA) (Morabito, 2015). Some traditional data processing algorithms are inadequate to use with these large data sets due to its complexity and length. BDA has to deal with a few challenges to be developed, such as analysis, capture, share, storage or visualization of the data to understand the different relationships among them. Thus, new algorithms are being developed to analyze these data such as Machine Learning (ML) ones. These type of algorithms are oriented to learn from the data and build predictions based on them (Kohavi and Provost, 1998). Normally these algorithms consist of building a model from example data sets and try to generalize them in order to make data-driven predictions or decisions, rather than following programmed steps externally (Bishop, 2006). There are many approaches of ML, some algorithms used are: decision tree learning, Radial Basis Function (RBF), Support Vector Machine (SVM), Principal Components Analysis (PCA), least-squares optimization, Artificial Neural Network (ANN) and Genetic Algorithm (GA), among others (Marsland, 2009). Most of the algorithms come from the Artificial Intelligence (AI) field, so that ML is considered as a branch of AI in charge of handling huge amount of data and extracting features from them (Blum and Langley, 1997).

But how can these AI algorithms help grids around the world? And why are these algorithms particularly interesting to the electrical sector? As described in Chapter 2, the evolution of the electrical grids towards a SG is becoming real. The installation of an Information and Communications Technology (ICT) infrastructure is generating lots of data which are necessary to be processed by grid operators, in order to complete their tasks in a more efficient way. Data processing must be fast enough to make instant decisions and effectively use them to take advantage of all the benefits that report the SG (see Section 2.3). However, the grid is currently a centralized system that makes difficult to process the vast amount of data generated per minute from smart metering. Thus, it will be smart to use some of these techniques, alleviating the data load from the central node. In addition, the inclusion of these algorithms makes feasible the decentralization of the system, leaving the decision-making to different nodes (e.g. substations) within the grid. If decentralization was taken to the end of the electricity system (i.e. users who manage their data, process it and take decisions in order to achieve a common objective), all the parties within the grid would benefit from it, being able to overcome all actual problems. In this scenario, nodes could enhance the system operation, being able to manage their power flows inside the grid.

This Thesis defines a decentralized algorithm to enhance the grid status through the management of the power flows inside the users facilities. The use of one of the classic and more famous AI algorithms is proposed, the ANN, in order to fulfill the objective of enhancing the grid operation by providing decisions to its members. Nowadays, ANNs are enjoying a growing popularity in applications because of their features, among which stand out its computing power, decentralized nature and generalization. In addition, through the development of ML techniques and improving computer hardware, neural structures are increasingly moving towards a greater resemblance to the brain (Eliasmith et al., 2012) or their use in deep learning (Bengio, 2009). In recent years, also biological neural networks have experienced intense development within the field of the neuroscience (Kandel et al., 2013) and the development of the technology has made possible to carry out deeper studies.

But why are these algorithms chosen? And why are they suitable to be applied in the management of a grid? In order to answer these questions, it is necessary to think in the resemblance between biological neural networks and the electrical ones. Both are large networks that are made up by a lattice of a huge amount of nodes interconnected. Electricity is present in both networks as the mean to stimulate the nodes, one to produce a reaction in the neuron and the other one to supply the demand and produce work. The use of ANNs within the grid is very suitable because of its distributed properties, adaptability, generalization, redundancy, etc. Properties which are quite similar to the ones that the future SG will provide to conventional power systems. ANNs also offers the possibility to establish different hierarchies and subdivisions to group the nodes, making easy to divide the problem into smaller subtasks. The modularity of the ANNs offer great possibilities to grid operators, by dividing it into portions that are easier to manage by adding more security to the system. Moreover, in a scenario in which users become "prosumers" (consumers plus generators), the use of ANNs is essential. ANNs provide users with tools to control their power flows in order to coordinate with the rest of users like the neurons in the brain, and to obtain collective benefits for the best grid operation.

The reminder of this Chapter is as follows. A brief historical review is presented in Section 3.1. Then, what are ANNs and their benefits are explained in Section 3.1.1. The actual tendencies are described in Section 3.1.2. After that, some architecture and types of ANN are introduced in Section 3.2. The type of ANN selected for its application in this Thesis is explained in Section 3.3. Section 3.4 introduces different applications in which ANNs are typically used. Section 3.4.1 will cover the different implementations of ANN in electrical applications. Finally, Section 3.5 explains different forms to train ANNs for the application they were designed for.

3.1 Historical Review

Throughout history, the study of the brain and the nervous system, as well as how humans can learn and perform tasks, has been the subject of many studies and scholars. In ancient Greek, the first studies on the ideas and the thought were developed by the philosophers Plato (428–348 B.C.) and Aristotle (384–322 B.C.). They together with other ancient Greek philosophers speculated about the causes of behavior and the relation of the mind to the brain. On the other hand, there was no physical interpretation of how the brain and the nervous system were, until the physician Galen in the 2nd century elaborated his theory. In Galen's theory, the nervous system consisted of a lattice of tissue interconnected from the brain to the nerves in the periphery of the body that conveyed fluid secreted by the brain and the spinal cord (Kandel et al., 2013). However, it was not until the 19th century that the actual knowledge of the brain structure was discovered by Ramón y Cajal. He discovered that the nervous system consists of a network of discrete cells, which would be called neurons, rather than a continuous web of elements (Ramón y Cajal, 1909). During his research, Ramón y Cajal developed a great part of the concepts and evidences for the neuron doctrine, which consists of the principle that the building blocks of the brain are the neurons and they are also the signaling elements of the nervous system (Kandel et al., 2013).

At the same time, the belief about artificial beings was also developed since antiquity through the mechanical or "formal" reasoning by philosophers and mathematicians. The development of the field of logic led to inventions in this area such as the programmable digital electronic computer. This development was based on the ideas of different mathematicians. One of these mathematicians was Alan Turing, who in 1936 developed a machine called the Turing's machine that was able to manipulate symbols and simulate any conceivable act of mathematical deduction (Berlinski, 2000). This fact, together with the development of other areas such as information theory, neurology and cybernetics allowed the development of a convergence study of the previous areas known as Artificial Intelligence (AI). This inspired different researchers to consider the possibility of building an artificial brain and the race of Artificial Neural Networks (ANNs) began.

The concept of developing an artificial brain began its first steps with the creation of the neurocomputation in 1943 by Warren S. McCulloch and Walter Pitts. The first one was a neuroanatomist and psychiatrist, while the second one was a mathematician. They described that any neural network could compute any arithmetic or logic function, joining the fields of neurophysiology and mathematical logic (McCulloch and Pitts, 1943). They proved that a neural network, consisting of a number of simple processing units with proper connections set among them, would perform any of those functions. This landmark means the beginning of the ANN and the birth of AI through neural computation. The influence of McCulloch and Pitts (1943) was such that it inspired von Neumann and Wiener research in von Neumann (1951) and Wiener (1948), respectively.

Next milestone in the development of ANN is the publication of the book of Donald Hebb, entitled *The Organization of Behavior* (Hebb, 1949). In his book, Hebb explained the learning process from a psychological perspective and the psychological conditioning which is ubiquitous as a property of neurons. The novelty of his work consisted of the formulation of a learning rule that modifies the synapses of neurons. He proposed that the connections of the brain are changing continuously as different tasks are learned and neural assemblies are created in those changes. The famous Hebb's rule was introduced in this book and stated that the variance of a synapse between two neurons is increased by the activation of one or by the other across its union (Hebb, 1949). This rule will influence many researchers from that time until today to elaborate learning synaptic algorithms of ANNs.

During the 50s some interesting events happened that continue the development of the ANN such as the construction of the first neurocomputer by Marvin Minsky in 1951 called Stochastic Neural Analog Reinforcement Computer (SNARC) (Zurada, 1992). SNARC was technically a success, but it did not do any information processing function. In addition, different studies during this decade contribute to posterior achievements: adaptive behavior, nonlinear adaptive filters or works on associative memories (Haykin, 2009). Also, an important event took part in this decade, the Dartmouth Conference of 1956.The conference consisted of trying to apply learning aspects to machines and simulate it. This event gave birth to the AI field as it is known today.

By the end of the 50s, particularly in 1958, the first successful neurocomputer was developed by Rossenblatt. The perceptron was born and consisted of a supervised learning method to the pattern recognition problem (Rosenblatt, 1958). His work was also known as the perceptron convergence theorem and it was himself the first one to prove its validity. The perceptron was the first ANN able to generalize, i.e. after learning a set of patterns, it could recognize similar ones that were never used during the learning phase. Then, Widrow and Hoff developed a new algorithm called Least Mean Squares (LMS), which was the pillar to build the ADAptative LINear Elements (ADALINE) (Widrow and Hoff, 1960). Nowadays, this algorithm is still in use, unlike the perceptron learning law. The ADALINE was applied in a real problem as adaptive filter to eliminate the echoes in telephonic lines. It also inspired the use of other more complex structures, such as the Multiple ADALINE, and other stochastic learning algorithms to be applied to ANNs.

With these events, it seemed in the 60s that ANNs were the solution to all the problems, prove of that is the wide research done in the field and the number of applications in which ANNs were employed. However, this glorious era came to an end with Minsky and Papert (1969). Perceptron limitations were highlighted in this book. They proved that single layer perceptrons cannot compute non linear functions and doubted that multilayer perceptrons or any other structure could solve this problem. They left the impression that the research in this field was a dead end. In addition, another reason that favored the abandon of many researchers was the lack of technological devices to carry out experiments.

During the 70s, the research of ANNs seemed to be dead, but some psychologists and neuroscientists continued the research in the field, focused on adaptive signal processing, pattern recognition and biological modeling. In (Amari, 1972), the learning in ANNs with thresholds was studied and he developed a stochastic gradient method. Stephen Grossberg developed different ANN structures and theories about adaptive resonance networks (Grossberg, 1974). These theories consisted of a new principle of self-organization with a recognition layer and a generative layer. At the same time, the inception of what later became one of the widespread algorithms, the backpropagation algorithm, was developed by Werbos (1974). In his thesis, Paul Werbos applied successfully an efficient reverse-mode gradient computation that was applied to general networks. Associative memory research was also developed in this decade and the studies in self-organization continued in Kohonen (1977). Kunihiko Fukushima developed another neural structure known as neocognitron (Fukushima, 1980). This structure consisted of a model that emulates the retinal image of the neurons trying to solve problems of visual pattern recognition. These researchers and many others in this decade were the people who prepared the ANNs to their next step and help to develop the base of the new achievements found in the next years.

Then in the 80s a renaissance era began for the study of ANN. Hopfield (1982) supposed a turning point in between the dormancy of the 70s and the beginning of a renewed enthusiasm on ANN. The work of Hopfield consisted in the study of recurrent structures with symmetric connections using the idea of an energy function to understand the way in which this structure computes the information. Hopfield (1982) encouraged hundreds of scientists, mathematicians, and technologists to rejoin the emerging field of neural networks. Another milestone that influenced this era was the publication of Rumelhart and McClelland (1986), which used the backpropagation algorithm to train multilayer perceptrons. Formally, the backpropagation algorithm was formulated at mid 80s, but it was first described in the work of Werbos (1974). These two publications led to the resurgence of the ANN field to explore neurocomputation. In the following years, the ANN increased significantly its expansion with the foundation of the International Neural Network Society, a journal to publish the research work in this field, IEEE Transactions on Neural Networks and different conferences such as the IEEE International Conference on Neural Networks.

This era will support the ANN expansion in the upcoming years of the 90s. The research on structures, learning algorithms and self-organizing methods will continue to develop, such as the Radial Basis Function (RBF) that consisted of a feedforward network alternative to perceptrons, or the Support Vector Machine (SVM) consisting on a supervised learning network to solve pattern recognition and regression problems. However, there was also an expansion to use these algorithms in other fields such as control, prediction and finance, taking advantage of the many qualities that such structures have. Also, during this time a concern about the dynamics of ANNs rose among researchers. Freeman (1995) tried to explain the existence of chaotic behaviors in neural structures and the emergence of self-organized patterns in populations of neurons. Furthermore, the advance of the technology in this decade allowed building specific hardware to compute the operation of an ANN. Thus, the operation of the

ANNs is done faster in the dedicated hardware providing margins to reduce system costs. An introduction of hardware neural networks is described in Goser (1996). Nowadays, the expansion of ANNs continues in old and new areas of applications because of several reasons that are introduced in next Section.

3.1.1 Why using Artificial Neural Networks in electrical systems

The use of ANNs is recently becoming more popular due to several reasons. But, why is it interesting to apply them in electrical systems to manage power flows? Before explaining why are so useful, it is necessary to understand what an ANN is and which is the operation that it performs. ANNs are algorithms biologically inspired in the brain, which are able to process the information in a distributed and parallel way. According to Haykin (2009), a neural network is an adaptive machine in the following way:

> "A neural network is a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects:
>
> - Knowledge is acquired by the network from its environment through a learning process.
> - Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge."

So, each node of the ANN receives the information from other nodes, processes it and elaborates its response. However, it is necessary to establish the right connections among the different nodes of which the ANN is composed. In general, a training period for the ANN is required in order to learn the function that it will perform. During this period the learning algorithm is in charge of modifying the different connections of the neurons and establishing the right relationships among them. There are different algorithms used in this training process and they will be introduced in Section 3.5.

The inspiration to create ANNs resides in the power of the brain to process information and extract relationships from the output world to the internal nervous system, taking a decision and carrying it out. However, it is important to note that ANNs are only inspired by the biological nervous system, and their operation is far from the actual behavior of the biological neurons (which are nowadays not completely understood). The nervous system consisted of discrete individual cells called neurons (Ramón y Cajal, 1909), which are able to process external stimulus and generate a response from them. The nervous system is divided in three main parts, receptors, the brain and effectors (Arbib, 2003). *Receptors* are the part of the system responsible for transforming the external stimuli captured by the nerve endings of the body into electrical impulses that the brain can interpret. The *brain* is the central part of the nervous system, and it is continuously receiving information from other parts of the nervous system. It is in charge of processing the information from receptors, and generates a response that it is transmitted to the ends of the nervous system. *Effectors* are responsible to transform the electrical signals from the brain into a response to the external world. However, the electrical stimuli flows in the nervous system not only in one direction (from receptors to the brain and from the brain to effectors), there is also a feedback from effectors to receptors that helps the brain to process the information from the outside and elaborate more complex responses.

The brain is the central processor of the nervous system and it is composed of neurons. The operation that a neuron can perform is carried out in the range of the milliseconds, meanwhile silicon chips can perform operations in nanoseconds. But, with the massive amount of neurons that the brain contains and the connections among them, the operation rate is reduced by processing information in parallel. The brain of an adult can contain approximately 120 thousand million of neurons which suppose around a 100 trillion of connections among them (Herculano-Houzel, 2009). There are different kinds of neurons inside the nervous system, which accomplish different tasks such as motor, connect muscles to the nervous system, sensory, receive signals from sensor organs, interneuron, to connect different neurons, etc. (Kandel et al., 2013). Apart from doing complex tasks, another important brain feature is how efficient its structure is. The energy consumption per operation in the brain is billions times higher in comparison to a computer processor (Haykin, 2009). These are some reasons that inspired the implementation of artificial structures. They try to imitate the brain structure in order to solve some complex tasks that the brain solves easily, such as pattern recognition, knowledge representation, task decomposition, etc.

Biological neurons can have different forms depending on the function they perform, but in general they all have three main parts. Figure 3.1(a) shows the three parts in which the neuron is divided. In general, the neuron has *dendrites*, which serve as communication channels for the input signals. A little processing information can be carried out in the dendritic inputs, but the main processing is done in the body of the neuron called *soma*, (see Figure 3.1(a)). It can also receive directly the information from the outside without coming from the dendrites. Finally, the response is transmitted using the *axon*. Communication among neurons is done through a structure called synapse, which passes a chemical or electrical signal to another neuron. The most general type of synapse is the chemical one and consists of passing neurotransmitters from the previous or presynaptic neuron to the next or postsynaptic neuron. Neurotransmitters initiate an electrical response in the postsynaptic neuron that may excite or inhibit it. The other possible synapse is electrical and connects a neuron directly to the other through a channel called *gap junction*, in which an electric current is passed, causing voltage changes. Then, inside the postsynaptic neuron, the electric potential generated (it could be positive, excitatory, or negative, inhibitory) is accumulated. When the potential generated surpasses a threshold, an action potential or spike is produced and it is propagated through the axon releasing neurotransmitters or an electric current at the end of the neuron.

Now, that the function of a biological neuron has been presented, it is easy to explain how the artificial model is inspired on them. The artificial neuron is represented in Figure 3.1(b). This model represents a generic artificial neuron and the following elements are identified (Haykin, 2009):

- *Inputs* (x_j). They represent the set of outputs of the presynaptic neurons. Both inputs and outputs may be binaries or continuous depending on the application and the model used.
- *Synaptic weight* (w_{ij}). They symbolize the synaptic connections among neurons and their values represent the interaction between the presynaptic jth neuron and the postsynaptic ith neuron.
- *Propagation rule* ($f(w_{ij},x_j)$). This operation consists of grouping the inputs making use of the relationship between the previous jth neuron and the ith neuron through the synaptic weights w_{ij} and obtaining what is called the postsynaptic potential (ν_i). The most common propagation rule used is a linear operation and consists of a weighted sum of x_j and w_{ij}, see Equation 3.1.

$$\nu_i = \sum_j w_{ij} \cdot x_j \tag{3.1}$$

  The inhibition or excitation of the neuron is the result of this operation. Thus, in the case of positive inputs and negative weights, the neuron will tend to be inhibited, while if the result is positive, it will tend to excite it. There are other propagation rules such as the euclidean distance (see Equation 3.2).

$$\nu_i = \sqrt{\sum_j (x_j - w_{ij})^2} \tag{3.2}$$

  In this case, when the difference of inputs and weights is small, the neuron will be inhibited. While if the difference is higher, the neuron will be excited.

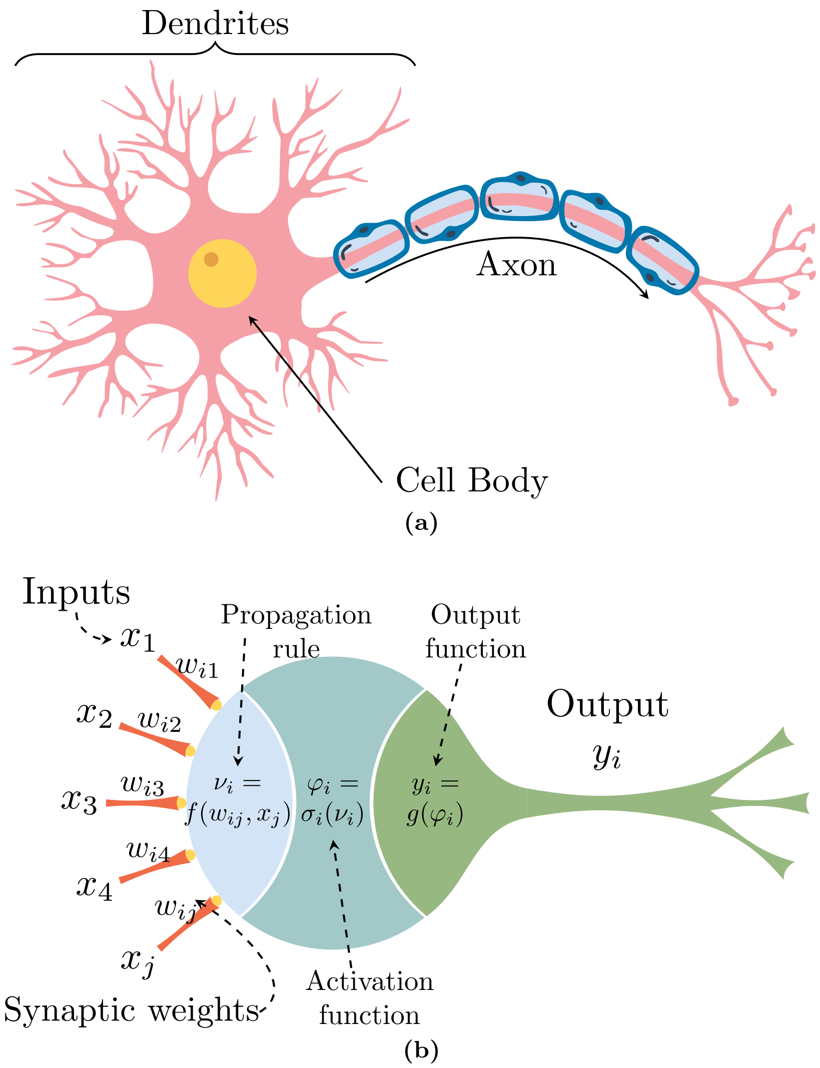


Figure 3.1: Neuron representation: (a) biological and (b) artificial.

- *Activation function* ($\sigma_i(\cdot)$). It provides the actual value of the neuron activation. Biologically this value depends on the previous activation value and the actual ν_i because the neuron fires a spike when the action potential surpasses a threshold. However, classical artificial neurons provide a limited output and in general only depend on the actual activation value and not previous ones. In order to represent the threshold of the neuron, there is another parameter, called *bias* (θ_i). If $\nu_i \geq \theta_i$, the neuron fires, otherwise there is not activation of the neuron. Thus, the effect of the θ_i increases or decreases the activation function result depending on its positive or negative value. Equation 3.3 shows the general form of the activation function.

$$\varphi_i = \sigma(\nu_i + \theta_i) \tag{3.3}$$

$\sigma_i(\cdot)$ can be of different forms as shown in Table 3.1, but normally the activation function is a deterministic function, monotonically increasing and continuous. Moreover, neuron stochastic models use a probabilistic activation function. The election of the activation function is normally given by the application. θ_i is normally added together with the inputs in the propagation rule.

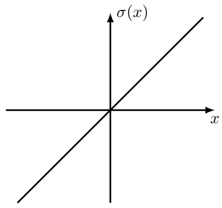
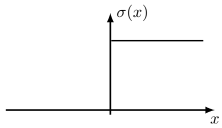
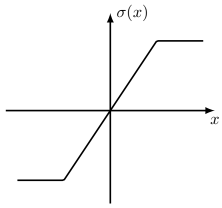
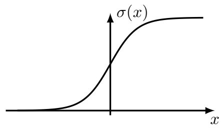
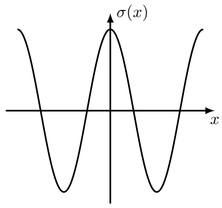
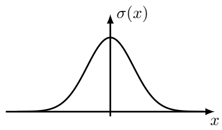
Name	Function	Range	Graphic Representation
Identity	$\sigma(x) = x$	$(-\infty, +\infty)$	
Step	$\sigma(x) = sign(x)$ $\sigma(x) = H(x)$	$[-1, 1]$ $[0, 1]$	
Piecewise	$\sigma(x) = \begin{cases} 1 & \text{if } x < -l, \\ x & \text{if } -l \geq x \leq +l, \\ 1 & \text{if } x > +l. \end{cases}$	$[-1, 1]$	
Sigmoid	$\sigma(x) = \frac{1}{1+e^{-x}}$ $\sigma(x) = tanh(x)$	$[0, 1]$ $[-1, 1]$	
Sinusoidal	$\sigma(x) = A \cdot \sin(\omega \cdot x + \phi)$	$[-1, +1]$	
Gaussian	$\sigma(x) = A \cdot e^{-\frac{(x-b)^2}{2 \cdot c^2}}$	$[0, 1]$	

Table 3.1: Example of different $\sigma_i(\cdot)$.

- *Output function* $(g(\varphi_i))$. This function provides the output of the neuron and its value depends on the activation. $g(\varphi_i)$ tries to emulate the propagation of electrical signals along the axon. Usually the identity function is used, so that the output of the artificial neuron is the value of the activation.

To sum up, the ANN operation is a composition of different functions, that tries to emulate the behavior of the biological neuron with several inputs and an output. The complete operation is described as follows:

$$y_i = g(\sigma(f(w_{ij}, x_j) + \theta_i)) \tag{3.4}$$

In this Thesis, the neuron model used is the one of Figure 3.2. The propagation rule consists of the weighted sum of inputs and synaptic weights plus the bias. Then, the activation function used is of sigmoid type and the output function is the identity. The mathematical expression is described in Equation 3.5.

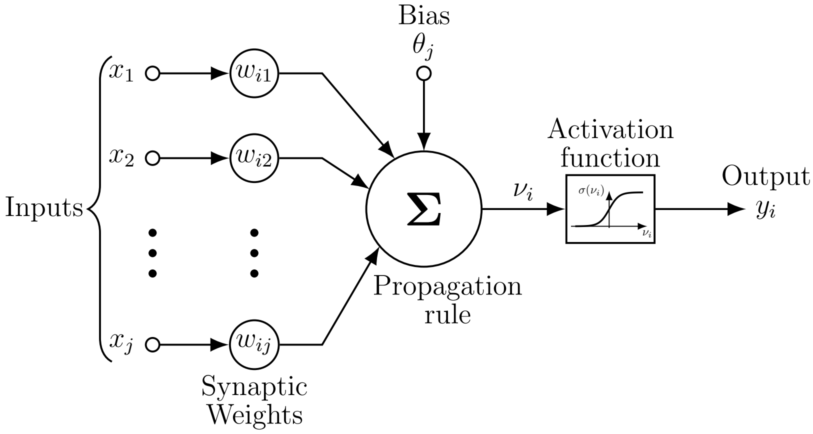


Figure 3.2: Artificial neuron model used.

$$y_i = \sigma\left(\sum_j x_j \cdot w_{ij} + \theta_i\right) = \sigma(\boldsymbol{x} \cdot \boldsymbol{w}_i + \theta_i) = \frac{1}{1 + e^{-(\boldsymbol{x} \cdot \boldsymbol{w}_i + \theta_i)}} \tag{3.5}$$

These neurons are gathered together to form different architectures conforming an ANN, designed in order to solve different problems. The ANN potential resides in its distributed architecture and ability to modify it through a training process to solve a task. These properties make easy to find a solution to complex problems that will required high efforts to solve it. Moreover, ANNs can process information never presented before even in their training phase thanks to their generalization ability. However, ANNs present other features such as the ones described as follows (Haykin, 2009; Jain et al., 1996):

- *Distributivity.* The architecture of an ANN consists of a group of neurons with connections among them. Therefore, the information is shared among all the units of the structure and the weights that connect them. This property allows the ANN to process the information in parallel and obtain a solution to the problem for which it was designed.

- *Redundancy.* The structure of the network possesses more than one way from the input to the output in order to reach it. The idea to establish redundant paths is to extract more easily features from the data and increment the robustness of the structure.

- *Learning.* ANNs learn from a training process in which the network modifies its weights based on some learning rule from the input data. Sometimes the network will learn from some outputs associated with the input (supervised learning), other times it will extract common features of the data and established different categories (non-supervised learning), and finally, there are times in which the weights are modified based on the environment and the actions that the network performs (reinforcement learning).

- *Adaptivity.* Neural networks possess a property to modify their synaptic weights due to changes in the environment that surround them. An ANN trained to solve a specific problem can modify their targets and learn from a new source of data to solve a new situation that is changing. Thus, their behavior responds to external stimuli modifying its response through changes in their structure. Moreover, when an ANN is operating in a non-stationary environment, the ANN itself is able to modify their synaptic weights in real time, thanks to learning algorithms such as the Hebb rule. In general, the more adaptive is a system, the more robust its performance will be, while stability is ensured.

- *Generalization.* Once the network is trained with a dataset, the network is normally executed with more data that was not presented in first place to the network during the training. If the ANN was able to extract the main features of the dataset, it will be able to generalize and obtain a right answer for other datasets. Otherwise, the ANN would not be able to generalize and obtain the information from the new data.

- *Non linear operation.* A neuron could be linear or nonlinear depending on the $\sigma_i(\cdot)$ selected. In turn, the ANN, built with these neurons, may inherit the same nature of the neurons which form part of it. In addition, the nonlinearity is distributed through the whole network. The importance of this feature is that an ANN can be used to solve nonlinear problems, such as the prediction of electricity generation and demand, or simulate the dynamics of a complex system like the grid.

- *Input-output mapping.* Some ANNs are able to learn through the environment in a training stage that consists of modifying the synaptic weights. This modification varies the weights in order to obtain the right output that it is also provided as part of the training. Each example is composed of the tuple of inputs and outputs, arranged randomly. So, the ANN is able to extract a relationship between the outputs and inputs presented during the training. In this way, when the network is presented with a new input never trained, it has to be able to solve the problem with a right answer.

- *Multi objective output.* The response of an ANN can be of different nature, being able to obtain different information at the same time from one input. For example, in a pattern recognition application, an ANN can be designed to select a pattern and at the same time the confidence in the made selection. So, an ANN can be built with the outputs needed to develop different tasks.

- *Contextual information.* The knowledge of the ANN is represented by different elements that form its structure and their activation. Each neuron within the ANN is affected by the overall activity of the other neurons.

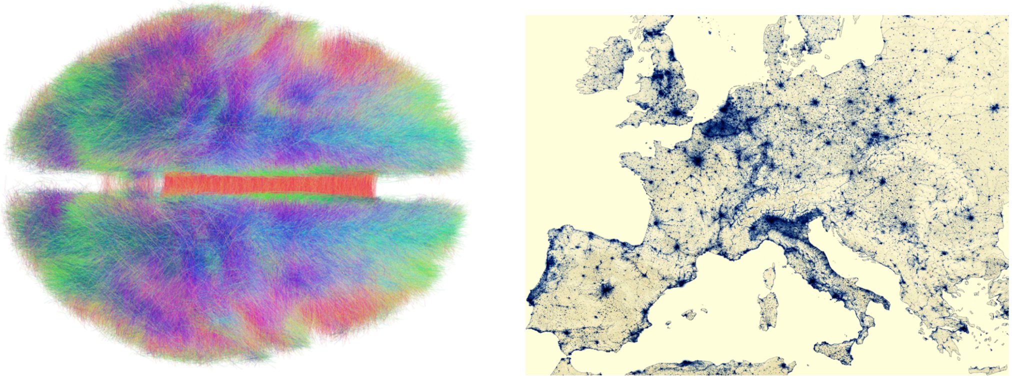
- *Fault tolerance.* This is another built-in property of the ANN because of the structure distributivity. The knowledge is stored through the whole structure. If a connection fails, the information is rerouted to other parts of the network and it can continue giving a response. In spite of loosing a part of the structure, the network structure is robust.

- *Task decomposition.* Thanks to the information distributivity inside the network, it is easy to build small structures that solve parts of the problem. Then, the result of each part of the problem is integrated into a more complex response to solve the task to which it was built in first place. This property makes possible the simplification of the problem and also simplifies the structure of the network.

- *Analysis and design uniformity.* In spite of the different applications and structures of the ANNs, there exist some conventions in their design. For example, the notation is the same for different applications. All ANNs are formed by neurons, so that different theories and learning algorithms can be shared for different applications.

- *Low energy consumption.* Because of its massive parallelism computing capacity, ANNs are very efficient and the electricity consumed in each operation by unit is really small compared to other computation structures. This property facilitates their integration in those applications in which energy is a problem, but a high processing capacity is required.

ANNs present lots of benefits that make them useful for almost any application. However, they also present some difficulties related to its structure. One of the



(a) **(b)**

Figure 3.3: Physical appearance of the connections inside: (a) the human brain and (b) the grid.

problems is related with the training. In order to correctly train an ANN, it is necessary to have a diversity of data that covers all the possible situations. Moreover, it also needed to adjust the different parameters of the training process properly to optimize the whole process. As more neurons are added, it is more complicated to train the network and the time that it takes is larger (curse of dimensionality (Bellman, 1957)). Processing and storage resources also grow with larger network structures, affecting their effectiveness and performance.

Nowadays, ANNs have recovered some popularity based on some factors, such as the technology advance and the need of processing huge amount of data as quickly as possible. In this Thesis, the use of ANNs to manage the power flows is not so obvious. However, there exist some characteristics in common between a neural network and the grid. For example, Figure 3.3 shows the physical appearance of the connections in both structures. Both structures possess lots of different and numerous nodes connected among them and a hierarchy is established to manage all of them. However, the grid is mainly a centralized entity with high robustness that it is not as efficient and robust as the brain. With the arrival of the SG, a greater distributed structure of the grid will be deployed and distributed approaches such as ANNs would help the management and integration of the different systems that will compose it. In an ANN, the information can be bidirectional, another desirable property in the SG. Thus, ANNs are used to enhance the grid operation through the reduction of the demand variability and integrating Distributed Energy Resources (DER) by using its built-in properties.

3.1.2 Tendencies in Neural Networks

As mentioned before, ANNs are becoming fashionable at present due to some factors related with the advance of technologies, tools and programming of different areas. New applications in which massive amount of data are available make necessary fast algorithms to extract the information required. At the same time, the data ubiquity and the places to take actions make very interesting the application of techniques such as the ANNs. There are three main fields in which ANNs are having a great push for its development: neuroscience, electronics and algorithmics.

- *Neuroscience.* The ANN is bioinspired in the brain. Neuroscientists use ANN as a tool for simulating neurobiological phenomena in computers. In this way, a part of neuroscience, called computational neuroscience, is actively developing the study of the nervous system from information processing structures that make it up. Computational neuroscience uses computer simulations and theoretical models based on the neuron records to study the functionality of each part of the nervous system (Dayan and Abbott, 2001). There exist different models that represent different aspects of the neural systems and the degree of abstraction is different for each of them depending on the brain area studied. In general, the neuron model used in computational neuroscience is a spiking neuron. The spiking model consists of a more realistic biological model of a neuron. In this case, the time is added to the other components of previous models, activation state and synaptic weights. This model considers the electric current that circulates inside the neuron and the potential that it creates, modeled as a differential equation. Thus, a spiking neuron only fires when its activation potential reaches a specific value, and then a relaxation period of the activation is necessary for the neuron to fire again. The neuron, which has fired a spike, generates a signal to the rest of neurons that are connected to it, increasing or decreasing their potential depending on the signal (Dayan and Abbott, 2001).

The spikes train of a population of neurons is then encoded so that the information is easier to interpret. And with this information the model is built to simulate the concrete part of the brain that the neuroscientists want to analyze. Therefore, the idea of these models is to convert the information of the external world of the brain to the neural space through the encoding of the spikes train generated by the stimuli of a neuron population. Then, a series of transformations are done with the encoded spikes and finally translated out of the neural space by another population of neurons to understand from the outside of the neural model the actions that have been taken place inside.

One tool that is gaining prominence within the neural simulation scene is the Neural Engineering Framework (NEF). Its novelty resides in the characterization of population-temporal coding that combines population vectors with linear decoding of neural spikes. This framework is based on signal processing and statistical inference, providing a robust strategy and evaluating the function of a wide variety of specific neural circuits. NEF is currently being applied to sensory processing, motor control and cognitive function (Eliasmith and Anderson, 2003). This framework consists of three principles:

i) *Representation.* The representation principle consists of how the information is coded and decoded inside the neural space for the neuron population. The encoding of the input information is nonlinear while the decoding used in the output neuron population is linear.
ii) *Transformation.* Transformation of neural representations are functions of variables represented by the different populations of neurons. This transformation is applied by a weighted linear decoding into the output population.
iii) *Dynamics.* And finally, the last principle consists of applying control theory to simulate neural representation dynamics for the different neuron populations.

NEF provides an easy way to encode and decode populations of neurons, control theory as a way to simulate their dynamics, a general way to generate neural systems with analytical synaptic weights for the functionality desired and promotes the formulation of specific hypotheses about circuit functions and key design constraints (Eliasmith and Anderson, 2003).

With NEF, the first working computational simulation of the brain has been implemented in Eliasmith et al. (2012), and is known as Semantic Pointer Architecture Unified Network (SPAUN). SPAUN is able to carry out about 8 cognitive and non-cognitive tasks that can be integrated in a single large-scale spiking neuron model. It is able to develop the following tasks: recognition, drawing, reinforcement learning, counting, working memory, answering questions, create variables and reasoning. SPAUN is able to switch between the different tasks automatically without any manual change in the routines of the programs. Thus, it is a fixed model that integrates perception, cognition, and action across several tasks.

In the race to the brain simulation, there are other projects that try to emulate the brain functioning by simulating each of the different neurons of the brain and putting them together, so that they can do the same tasks as a human brain[1]. Another interesting project is the Human Connectome Project, which aims at studying the connections among neurons in the brain trying to elaborate an atlas of the different parts of the brain[2]. Both projects contribute to have a better understanding of how the brain works, but they are not implementing a functional model that completes tasks from an AI point of view such as SPAUN. As mentioned before, NEF applies control theory to understand the dynamics of the neural models. However, there is another aspect inside neuroscience that tries to directly study the neurons and the association between them as dynamical systems (Izhikevich, 2007). In this case the idea is to explain neural models through the dynamics of spiking neurons and observe the behavior that comes from their interactions to explain how a neural ensemble works. In Izhikevich (2007), the nonlinear dynamics systems of the brain are explained, beginning from a low dimensional neural space to a higher one.

- *Electronics*. The next field in which ANNs are having important advances is in hardware for their implementation and neuromorphic computers. This idea is gaining more followers because technology is mature enough to implement directly the ANN behavior at an affordable cost. In addition, the costs decrease as volume production increases, so it could be interesting to adopt a specific hardware in case the application volume is higher. Another advantage of hardware implementation is the increase of speed since the neural operation is dedicated. There are many classes of hardware used to implement ANNs (Misra and Saha, 2010):

  - *Digital* chips consist of specific hardware dedicated to emulate neurons, in general they use CMOS technology and store the weights in RAM, which allow a flexible design.
  - An *analog* neuron is emulated in hardware through resistors, charge-coupled devices, capacitors, etc. Thus, the hardware itself implements the neuron structure, being the design more rigid than digital one.
  - Field Programmable Gate Arrays (FPGAs) provide an effective way to program different neural structures through their reconfigurable structure in a very short time and with low costs. It provides software flexibility and online reconfiguration capabilities (Misra and Saha, 2010).
  - Fast Graphics Processing Unit (GPU)-based implementations of ANNs are also being used to emulate large structures because as FPGAs, they allow reconfiguring rapidly the structure and the computation capacity is higher than CPU implementations.
  - The chips that are currently having greater development are the *neuromorphic* ones. Neuromorphic refers to circuits that emulate closely the biological neural design. Almost all the processing is analog, but their outputs can be digital. An example of neuromorphic chip is the Neurogrid developed in Benjamin et al. (2014). The board is composed of 16 custom-designed chips, referred to as NeuroCores. NeuroCore analog part consisted of the emulation of 65536 neurons trying to maximize the energy efficiency. The emulated neurons are connected using digital circuitry designed to maximize spiking throughput.

- *Algorithmcs*. As technologies and models advance, they allow spreading the fields of application in which ANNs can actuate. This is reason why the ANN structure can be further complicated by adding more neurons and using more complex hierarchies with more elements. At present, ANNs are being used in Machine Learning (ML) algorithms as one of the principal techniques due to the nonlinearity operation and the possibility to establish easy relationships among

[1]Blue Brain Project official website: `http://bluebrain.epfl.ch/`

[2]Human Connectome official website: `http://humanconnectome.org/`

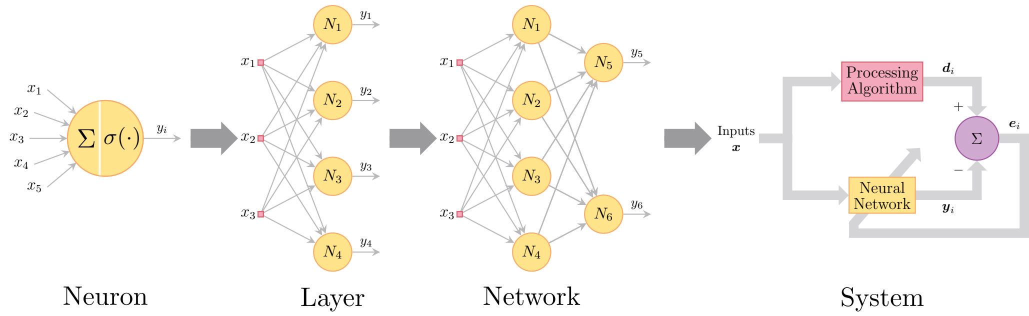


Figure 3.4: Different association of neurons to form different structures.

massive quantities of data. Another interesting ability for ML is elaborating general purpose predictions based on the data available, in which ANN stands out (Blum and Langley, 1997). In addition, the ANN feature extraction has been improved due to the increasing complexity of the network structure, being able to process not only lager datasets, but also fast enough to consider it as a promising algorithm to take into account.

As mentioned before, neuroscientist looks into ANNs as a research tool to implement neurobiological models. However, engineers also seek in neurobiology for new ideas to solve more complex problems. That is how Deep Learning (DL) arises in the actual scene of ANNs. This allows implementing network architectures that can be used for example in the vision recognition problems. In this case, the structure of DL-ANN consists of more neurons in the network, more connections among them and a deeper hierarchy with a greater resemblance to the human brain. One application in which these complex architectures are used is in computer vision in which convolutional ANNs are used (Krizhevsky et al., 2012). The idea of this structure is building different layers of neurons in which the feature extraction is done. A convolutional network is a type of ANN in which neurons are tiled in such a way that they respond to overlapping regions in the visual field. Google has also developed his own version of them, GoogLeNet (Szegedy et al., 2015). Furthermore, in Le et al. (2012), the authors uses a DL architecture that learned to recognize higher-level concepts only from watching unlabeled images. Other structure for feature extraction is the autoencoder, which tries to compress the information from a higher dimensional space to a smaller one that contains the main features from the previous one. Therefore, the complexity of new applications is evolving the concepts around ANNs and developing new models with major proximity to the theoretic goals of AI.

3.2 Architectures and Types

So far the principles, history and future of the ANNs have been introduced. In this Section, different types, architectures and structures are introduced. The basic building block of ANNs is the neuron and it is used as a small processing unit. A set of neurons is usually grouped forming layers which have its own functionality. Then, the different layers are also grouped together to form a global system, known as neural network. In addition, various neural networks can be associated to form a neural system, which is able to complete a specific task (see Figure 3.4). Both concepts could be applied in biological and artificial contexts. In an ANN, the different modules in which the system is divided do not have to be neural. Thus, an artificial system in which an ANN is taking part consists of inputs that go to the corresponding modules where the information is processed and then the system output is obtained to solve the task for which was built.

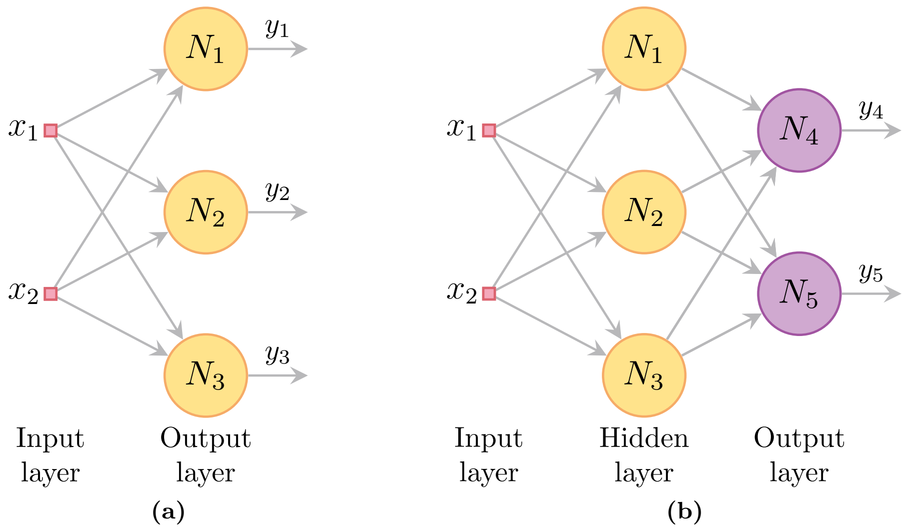


Figure 3.5: Different ANN structures attending to the number of layers: (a) single-layer neural network with an input and output layer and (b) multilayer neural network with an input, hidden and output layer.

Different models and architectures exist depending on the application and use of the ANN. The structure of an ANN is the physical layout of the network and the relationships of the different units that comprise the neural network are defined by different factors. One of these factors is the use of an algorithm to train the network with different data and establish the strength between neurons in order to fulfill the task for which is built. In this way, another factor that will determine the structure of the ANN is the application, because it will determine the size, relationship between nodes, flow of information, etc. Thus, one way to start a taxonomy of the different network structures consists of considering the number of layers that the architecture possesses (Haykin, 2009):

- *Single-layer*. In this type of networks, the neurons are organized in a single layer, although an input layer is located before this. The principal function of the input layer is to send the information to the neurons in the next layer which is also the output of the network (see Figure 3.5(a)). In the neuron layer the information of the previous layer is processed and the output is generated. However, the operations that the network can perform are restricted, and only linear operations can be represented with this type of network (Minsky and Papert, 1969). The number of neurons depend on the task that the ANN is going to fulfill.

- *Multi-layer*. This type of architecture is a generalization of the previous one and possesses one or more processing layers between the input and the output layer (see Figure 3.5(b)). The neurons in the intermediate layers are known as hidden neurons and their corresponding layers are denominated hidden layers. Its name is because the hidden layer is not accessible directly from the outside. In this architecture the output of the previous layers serves as inputs to the next ones. The information is processed in the hidden and output layers. The addition of hidden layers provides the network with the ability of obtaining statistics of higher order. This hidden layer also provides more connections that increase the dimensionality representation of the network. Thus, they are very useful when the number of inputs is large and it is required to extract useful features from the data. According to Cybenko (1989), a multilayer structure with

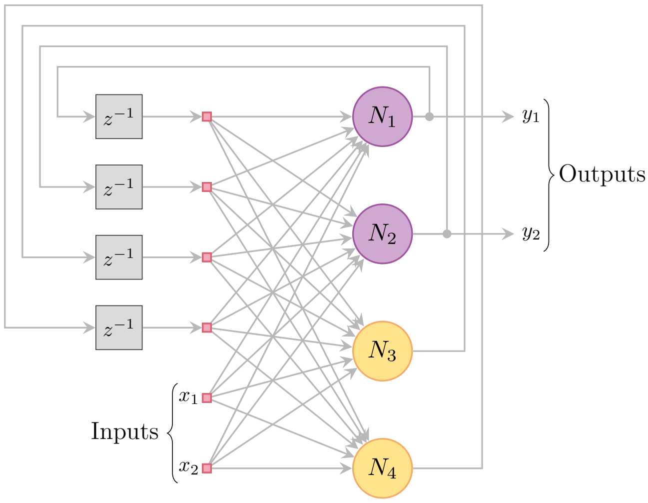


Figure 3.6: Feedback neural network structure.

sigmoid activation function is a universal approximator of continuous functions. Therefore, multilayer networks are frequently used due to these properties.

Another important characteristic related with the morphology of ANNs is the direction of the information flow inside the network. This feature is highly related with the synaptic connections between neurons and between layers. For example, in the networks of Figure 3.5, the information goes always from the previous layer to the next one, i.e. *feedforward* networks. The connections between different units do not make any cycle or loop, so that the information goes from the back to the front with no recurrence. However, there are another type of architectures in which the information does not only go from the back to front, but also there are some connections that allow the information going backwards, i.e. *feedback* networks. These type of networks are generally known as Recurrent Neural Network (RNN). Figure 3.6 shows the architecture of a RNN. There are loops between the different neurons of the network and the flow of information goes backwards through some feedback loops, composed by unit-delay elements (z^{-1}). These feedback loops could be between neurons of different layers, the same layer or themselves. RNNs result in a nonlinear dynamical behavior because of the feedbacks and make them particularly suitable for studying the dynamics of nonlinear systems.

So far, an ANN taxonomy was established based on the morphology and the aspect of the connections between neurons. However, they can be also classified according to the type of learning algorithm used. A learning algorithm consists of a set of rules that modify the relationships between neurons. These modifications consist of changes in the values of the free parameters of the network (w_{ij} and θ_i). The modifications are done so that the neural network is able to perform the task for which it was designed. In all learning schemes the ANN is fed with data from the environment which is the surrounding field of the application. There are different types of learning algorithms, the main ones are described as follows (Haykin, 2009):

- *Supervised.* This type of algorithm consists of teaching an input-output relationship to the ANN so that it can extract the connection between them through the modification of the free parameters (w_{ij} and θ_i) of the network. The ANN tries to learn from the environment a relationship through a teacher that provides the network with the input pattern ($\bar{\boldsymbol{x}}$) and the desired output ($\bar{\boldsymbol{d}}$). After the network is executed, its output ($\bar{\boldsymbol{y}}$) is compared to $\bar{\boldsymbol{d}}$ and the error between them is used to modify the free parameters of the network. The network tries to minimize the error difference through a function called cost function which will give the modifications of the network parameters. This process is

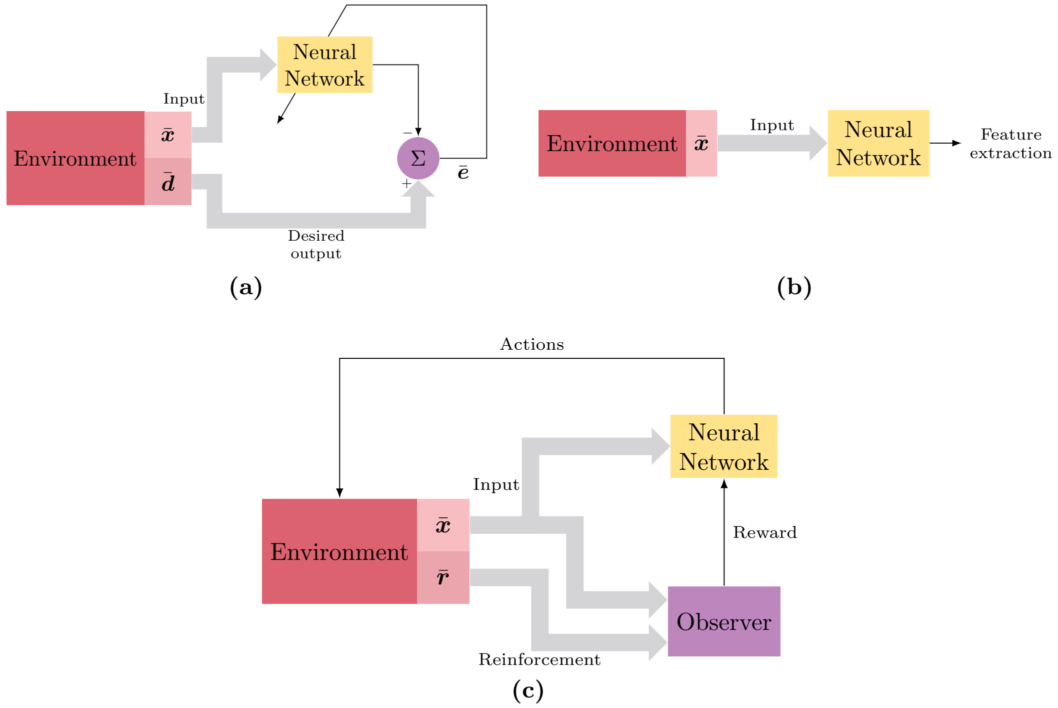


Figure 3.7: Learning schemes for ANN: (a) supervised learning, (b) unsupervised learning and (c) reinforcement learning.

repeated iteratively step-by-step with the aim of eventually making the neural network emulate the teacher; the emulation is presumed to be optimum in some statistical sense. Figure 3.7(a) represents a block diagram of this type of learning, it constitutes a closed-loop feedback system, but the unknown environment is outside the loop. Thus, the ANN, which has no knowledge of the environment a priori, is able to acquire this knowledge from the teacher and to store it in the synaptic weight as a long-term memory. Once the network has converged to the solution, the next phase is for the network to deal with the environment directly and to try to generalize its behavior with inputs not presented previously during the training. With the adequate set of input-output examples, the right minimization of the cost function and enough time to do the training, an ANN can approximate reasonably well the unknown input–output mapping through supervised learning.

- *Unsupervised.* In this case there is no teacher to learn from, as can be observed in Figure 3.7(b) and this is also one of the characteristics by which it is known as self-organized learning. The data coming from the external environment is unlabeled and there is not error or reward signal to evaluate the performance of the network. Hence, the input data is presented directly to the ANN and it will be able to extract some features from the data due to a task-independent measure. This measure is able to judge the quality of the network data representation related to what features the network has to learn. Then based on this measurement, the free parameters of the network are optimized in the way of this rule. Once the network is tuned for the statistical regularities of the input data, it develops the ability to form its own representations for encoding features of the input and to create new classes automatically. For example, a common rule used in unsupervised learning is a competitive strategy in which the neuron with higher output will be the output of the system (winner-takes-all). Unsupervised learning seeks to summarize and explain key features of the data which is related to data mining methods. This type of learning is closely related to the feature extraction from the data, trying to group them such as in clustering or in Blind Source Separation (BSS). One of the common rules is the methods of moments, which consists of estimating the free parameters of the network based on the moments of one or more random variables. Thus, these unknown parameters can be estimated given the moments.

- *Reinforcement.* This type of learning is more similar to the unsupervised learning, but the input-output mapping is done through the continuous interaction of the ANN with the environment. In this case, there is not an error signal and the desired output is not specified, but the network receives a performance index to minimize. This index only indicates to the network how well or badly its performance is with respect to the reinforcement signal from the environment. Figure 3.7(c) presents an example of reinforcement learning system, where a closed-loop feedback system is formed by an objective observer which receives two signals from the environment: the input ($\bar{\boldsymbol{x}}$) and the reinforcement ($\bar{\boldsymbol{r}}$) signals. Then, the objective observer elaborates a reward signal that indicates how well was the performance of the network to the last input and will be used to feed the network with the actual $\bar{\boldsymbol{x}}$. The network output will serve as influencing actions in the environment that will change it. In this process, the reinforcement signal arrives to the ANN one step after (delayed reinforcement), which eventually results in the generation of the heuristic reinforcement signal. The objective of this type of learning is to minimize a cost-to-go function, defined as a cumulative cost of actions over a set of steps rather than the immediate cost. Therefore, the network can find good solutions in the overall system behavior in order to discover these actions and feed them back to the environment. However, the delayed reinforcement learning presents two difficulties, i) the absence of desired outputs and ii) the delay on the reward signal makes the ANN to solve a *temporal credit assignment* problem. In spite of these difficulties, reinforcement learning is appealing because it gives some mechanisms to learn a task interacting with the environment.

- *Hybrid.* This type of learning consists of merging the previous types of learning in the same ANN. Thus, this type of learning could be applied in different layers of the same network or in different parts of a neural system with more than one neural network. The idea of this type of learning is combining the benefits of all of them and try to learn more complex tasks to solve faster the problem. However, it is difficult to apply because of the different variables to minimize their cost and the various optimization functions that modify the free parameters of the network. Yet, it is a powerful tool to learn multiple tasks in a given environment.

In Table 3.2, there is a classification of some of the most famous ANN models and learning algorithms. The classification has been done attending to the morphology of the models and the way of training the free parameters. The more important models included in Table 3.2 are described as follows:

- **Simple perceptron**. This architecture represents a binary classifier that can decide if a determined input corresponds to one class or another. It is a type of linear classifier that makes its classifications based on the linear prediction of a function combining w_{ij} with the feature vector. Rosenblatt (1958) describes the perceptron as a neuron theory and also as a neural network.

- **Multilayer perceptron**. This is a variant of the simple perceptron to which hidden layers have been added, so that it is able to solve problems that are not linearly separable. It was enunciated in Rosenblatt (1961), however there was not a learning algorithm able to train this type of networks until the arrival of the backpropagation algorithm.

- **ADAptative LINear Elements (ADALINE)**. It is a similar model of neuron as the perceptron and it has a linear response whose inputs are normally continuous. The network uses memistors, which are resistors with memory able

ANN models & learning algorithms			
Supervised		**Unsupervised**	
Feedforward	*Feedback*	*Feedforward*	*Feedback*
Perceptron	BSB	LAM/OLAM	ART
Adaline/Madeline	Fuzzy Cog. Map	Self-organizing map	Hopfield
Multilayer Perceptron	DTRNN/CTRNN	Neocognitron	BAM
Back Propagation	BP through time	PCA Network	Elman
Time-delay NN	Echo state Net		
Cascade Correlation			
Boltzman Machine			
Support Vector Machine			
LVQ			
GRNN			
CMAC			
Reinforcement		**Hybrid**	
Q-Learning		Radial basis function	
Temporal difference methods		Convolutional Networks	

Table 3.2: Models of Artificial Neural Network and learning algorithms.

to perform logic operations and store information. It consists of a weight, a bias and a summation function, but the main difference with the perceptron is that in the learning phase the weights are adjusted according to the weighted sum of the inputs (Widrow and Hoff, 1960). There is also a version with multiple ADALINEs called MADALINE.

- **Hopfield**. It is a type of RNN developed in Hopfield (1982), which serves as a content-addressable memory with binary threshold nodes. Inside the network, the input data is compared against each node of the network, then one of the neurons will fire and it will be the output of the network for that input. During the training this type of network guarantees the convergence to a local minimum, but this minimum can correspond to another pattern different from the one that the network has to learn. In addition, Hopfield proposed a way to understand the human memory (Hopfield, 1982).

- **Boltzman machine**. The Boltzman machine is another RNN with symmetric connections whose neurons make stochastic decisions about whether to be on or off (Hinton and Sejnowski, 1983). This type of network tries to extract features that represent complex regularities in the training data. The learning algorithm used is very simple but it is too slow when networks are too dense. They are generally used as feature detectors in classification problems. In this type of problems they represent a cost function in which the neuron with the highest activation would be the output of the network. It can be used also to reproduce a set of input vectors with high probability in learning problems (Hinton and Sejnowski, 1983).

- **Self-Organizing Map (SOM)**. It is also known as Kohonen network and it is a type of unsupervised learning neural network. The neurons are distributed regularly in the form of a grid, in general of two dimensions, whose purpose is to find the structure of the data inserted in the network. So, it makes a dimensional projection from the set of data given to the neural space. During the training, the data vectors are the input to each neuron and they are compared with the characteristic weight vector of each neuron. Then, the neuron with the lowest difference between its weight vector and the data vector is the winner and this neuron together with the surrounding neurons modify their weight vector. Thus, the neurons which are together in the SOM are more similar than the ones that are further away. SOMs also possess spatial information encoded in their structure forming an ordered mapping. They were applied in visualization applications, and applications related with associative memory such as pattern classification (Kohonen, 1977).

- **Adaptive Resonance Theory (ART)**. It is a model of ANN whose operation is based on the way in which the human brain processes information. The model consists basically of two layers, one of inputs to sense the environment and one of outputs, in which neurons compete with each other so that one is the response which inhibits the rest. There is a feedback between the output layer and the input one, so when an input is presented to the network if one of the output resonates with it, then it is associated with the neuron. The center of the cluster will be moved to this class for a better adaptation to the next input with the feedback between layers. In case that any neuron output is activated, the network could be saturated or a new class could be added to resonate with it. This type of network was invented to solve the problem of stability and plasticity during the learning phase (Grossberg, 1987). There are also different versions of the network.

- **Radial Basis Function (RBF)**. This type of networks consists normally of three layers: one input layer, a hidden layer with RBF and an output layer. A RBF is a function whose value depends on the distance from a point c, called center (Buhman, 2003). For example, one function normally used is the Gaussian function that perfectly meets this property. Hence, the idea of the RBF network is to use the RBF layer as a separation layer in which the different neurons behave as kernels, classifying the data and indicating to which neuron is closer. In addition, as RBFs are used to classify complex patterns, the hidden layer possesses more neurons than the input layer because a complex pattern recognition problem is more likely to be linearly separable in a high-dimensional nonlinear space.

- **Support Vector Machine (SVM)**. This is not really a type of network but rather a type of algorithm used for training them. It is used along RBFs or multilayer perceptrons with a single hidden layer. This type of algorithms are used to build networks to analyze data and solve problems related with the recognition of patterns, classification and regression analysis. SVM is used to build a hyperplane that serves as decision surface that will be maximized following the separation margin between examples. Therefore, the SVM can map new examples using the same space and predicted to which category they belong, depending on the side of the gap they fall on. As RBFs, SVMs use high-dimensional nonlinear space to separate the data through a non-linear classification (Cortes and Vapnik, 1995).

These are some of the most important architectures, also listed in Section 3.1. There exist many types of ANNs because there is not a unique established criterion to follow in order to build them. The idea is that a variety of ANNs can be built depending on various factors: number of neurons, number of layers, connections among them (neurons and layers), etc. Also, the function that develops the neurons can be configured using different activation functions or propagation rules. In addition, a more complex system can be built trying to gather together the operation of different ANNs. Thus, the idea is that each ANN solves part of a complex problem by dividing it in simpler different subtasks. Apart from reducing the model complexity, this modular approach will give other benefits such as simpler neural models, scalability, robustness and computational efficiency among others (Buhman, 1998).

There is nothing preset and any architecture can be built from the fundamental unit, the neuron, to the connections between them (neurons, layers or networks) and the form in which is trained. However, the application has to be present in the design of the architecture because there are some types of configurations that are more suitable for an application than others. Based on the application, a RNN architecture has been chosen in this Thesis because of its intrinsic properties and the ability to simulate a dynamical behavior. Section 3.3 will explain in further detail the RNN architecture used in this Thesis and the properties for which these architectures are so interesting.

3.3 Recurrent Neural Network

This type of ANN deserves special attention due to the properties that are provided by the feedback loops inside the network. The feedback in the network could be local, to the neuron itself or its surroundings, or global, to other layers or even other neural networks giving more connectivity options. Thus, the presence of loops makes possible that the information about the activation status of the neurons flows inside the network adding the property of short time memory to them. This representation is closer to biological neural networks in which recurrence is present in all parts of the nervous system. Adding short term memory to neural models makes a RNN able to do temporal processing and learn sequences, such as performing sequence recognition/reproduction or temporal association/prediction. On the other hand, it will increase the complexity of the network and there exist some theoretical and practical difficulties to implement in some applications so far.

The time variable is essential in the development of some tasks, such as vision, speech, signal processing, etc. Time representation could be done in a continuous or discrete way depending on the application. Including time in the performance of an ANN makes possible to follow statistical variations in non stationary processes, such as speech signal, the market stock or the electric demand of a country. So, the addition of this new variable makes possible to emulate the dynamics of a system. There are two forms of adding the temporal information to the neural architectures: i) implicitly, the network structure presents delays between synaptic connections, or ii) explicitly, the external information is sampled and different samples are passed to the network so the temporal information is in the input (Haykin, 2009). This last type was commonly used with static feedforward ANNs before the feasibility of implementing dynamic RNNs. Hence, the input has delayed samples of itself to introduce temporal information which will be stored in the synaptic connections of the neural network.

Thus, it is necessary that memory would be added for an ANN to be dynamic. Long term memory is built in the network through the learning process after which the synaptic weights store all the data information used during the training. On the other hand, short term memory is required in case the problem to solve has a temporal dimension. One way of building this is the inclusion of feedback loops inside the structure of the network, through time delays in the synaptic connections. The inclusion of time delays is motivated by the biological functioning of the brain and is related on how the information is processed in the brain (Braitenberg, 1986).

The dynamics of RNNs has been widely studied since the 70s (Wilson and Cowan, 1972). A dynamical environment which is constantly changing surrounds everything in daily life, so it is necessary to include dynamical properties in the artificial neural models to adapt them to the environment. RNNs can exhibit three different dynamical behaviors: i) convergent, ii) oscillatory and iii) chaotic, depending on the number of neurons and connections between them (Pasemann et al., 2003). However, these behaviors are not unique of RNNs, for example feedforward networks present convergence dynamics because of the characteristics of the applications in which they are used, such as content addressable memories or pattern recognition. On the other hand, oscillatory and chaotic dynamics only appear on RNNs, because of their neurons feedback (Dauce et al., 1998).

The interest in the oscillatory and chaotic behavior has been the subject of many studies in chemical, physical and biological systems (Tu, 2006; Goldbeter, 1997; Chay, 1981). The appeal of RNNs is centered in their dynamic characteristics. Although having the same characteristics that feedforward networks, RNNs are more complex and there is not a clear way to train them in spite of several learning algorithms. However, a RNN has the ability not only to be mathematically a dynamic system by itself, but also to approximate arbitrary dynamic systems with an arbitrary precision (Pearlmutter, 1995). To do that, it is necessary to train adequately the network in order to react and behave as such function according to the appropriate external stimuli. The most widely used learning algorithms to train RNNs are gradient descent algorithms, which correspond to the type of supervised learning (see Section 3.5). Furthermore, it is also possible for a RNN to learn the trajectory that describes the dynamic behavior of a system such as in Pearlmutter (1989). In this case, the network does not have any input and the connections are modified in order to follow the different states of the dynamic system.

Nevertheless, it is necessary to be careful with the addition of feedback to an ANN, because when applied improperly, they can produce harmful effects. The problem is that the application of feedback can cause a system a priori stable to become unstable. Therefore, it is also important to study the RNN stability to guarantee that the network is going to perform the task properly (see Section 3.3.1). The stability of a nonlinear dynamic system affects the whole system and implies some form of coordination between the individual parts of a system (Haykin, 2009). The stability study of a dynamical system makes possible to find solutions that avoid regions of instability and converge to a solution to the problem at hand.

Besides the approximation of arbitrary functions, RNNs are able to perform other tasks, such as content addressable memory, autoassociation, dynamic reconstruction of a chaotic process, etc. They are also able to map input sequences to output sequences, with or without a teacher. One of the reasons to use them is that RNNs are computationally more powerful and biologically more plausible than other adaptive approaches such as Hidden Markov Models (no continuous internal states), feedforward networks and SVMs (no internal states at all). The global feedback is a facilitator of the computational intelligence (Haykin, 2009). Thus, the network increases its capabilities to adapt to changes during the development of the application for which it was built in the first place. Typically, RNNs were applied to many areas such as control theory, robotics, pattern recognition, etc. But one area in which they are being used extensively is the time series forecasting due to its nonlinear dynamics. In addition, RNNs are preferably used rather than time series analysis because they make accurate predictions, are computationally faster, make iterative forecasts and deal with nonlinearity and non stationary input processes. Specifically, RNNs have found their place in predicting time series as Nonlinear Autoregressive Moving Average with eXogenous inputs (NARMAX) prediction models (Mandic and Chambers, 2001).

There exist different architectures also for RNNs depending on the loops of the network. Hopfield networks have been described in Section 3.2, other famous architectures are:

- *Fully recurrent network.* All the neurons are connected with each other, there exist connections from each neuron to the rest and itself. So there is not a structure defined as such, only a few neurons serve as inputs and outputs and the rest will be hidden neurons. This structure has a high configuration capacity, however it is also a problem to train the network when there is a large amount of neurons. The reason is that the number of parameters to train increases drastically with the addition of more neurons. Such networks fit well when the application is not enough defined or it is necessary to use brute force to solve the problem (Williams and Zipser, 1989).

- *Recurrent multilayer perceptron.* This type of networks consists of more than one layer which has feedback loops in the neurons of the same layer but there is not any recurrence with previous or posterior layers. Hence, the network has an input layer, one or more hidden layers and an output layer. The connection between them is done in a feedforward way. This type of network possesses the properties of the multilayer perceptrons plus the ability to incorporate temporal behavior through the feedback loops (Haykin, 2009).

- *Long Short Term Memory (LSTM) network.* This type of network was developed in Hochreiter and Schmidhuber (1997) to solve the problem found in the training of the layered networks. The network consists of an ANN to which some LSTM blocks have been added. A LSTM block consists of a processing unit that can remember a value for an arbitrary length of time, deciding when the input has to be remembered, when it should continue remembering it or forget the value, and when it should output the value. The property of storing an arbitrary value makes possible to work with long delays, and handle signals with a mix of low and high frequency components. This network outperforms in tasks related with classification, processing and predicting time series with lags of data of unknown size.

- *Echo State Network (ESN).* In this case the ESN is a specific type of RNNs that possesses a particular form, it has an input and output layers and between them a large reservoir of hidden units sparsely connected (typically 1 % connectivity). The connection and number of neurons in the hidden reservoir are assigned randomly and the only weights that need to be trained are the ones in the output layer. Thus, this neural network is appropriate at reproducing specific temporal patterns because of its nonlinearity properties (Jaeger, 2001).

- *Elman and Jordan networks.* This type of network follows a similar concept than the Hopfield networks. The idea consists of three layers (input, hidden and output) plus a context layer that stored the state of the hidden layer from the previous time step. The context units are connected only to the hidden layer, maintaining a copy of the previous values of the hidden units before the learning rule is applied. Thus, the network maintains a state, being able to perform tasks related with prediction better than feedforward networks (Elman, 1990). Jordan networks are similar to Elman ones, but with the peculiarity that the context units take their input from the output layer rather than the hidden layer (Jordan, 1986).

Some special architectures have been introduced above and only referred to some peculiarities related with their structure. However, there exist other architectures based on differential equations in order to model the neural behavior to external stimuli. These types of RNNs try to use the dynamical system theory to model biological neural networks that include the time as reference and feedback loops as part of the network structure, such as in the brain. There exist two different types of networks depending on the time basis used: i) discrete or ii) continuous. Discrete Time RNN (DTRNN) uses a difference equation to describe the neural model. The idea is that each time step the neuron activation evolves to a new value depending on the previous one (Pasemann, 1993). The mathematical representation of the DTRNN is shown in Equation 3.6.

$$y_i[k+1] = f_i(x_i[k], y_1[k], \dots, y_n[k]) = \sigma\left(\sum_{j=1}^{n} w_{ij} \cdot y_j[k] + x_i[k] + \theta_i\right) \tag{3.6}$$

where, y_i is the output of the ith neuron, $\sigma(\cdot)$ is the activation function, w_{ij} is the neural weight that connects the ith neuron with jth neuron, x_i is the input value of the ith neuron and θ_i is its bias value.

Equation 3.6 may be represented as an alternative vector form as:

$$\boldsymbol{y}[k+1] = f_\alpha(\boldsymbol{x}[k], \boldsymbol{y}[k]) = \varphi\left(\boldsymbol{W} \cdot \boldsymbol{y}[k] + \boldsymbol{x}[k] + \boldsymbol{\theta}\right) \tag{3.7}$$

where, $\boldsymbol{y}$ is the output vector of the neurons, $\boldsymbol{W}$ is the matrix of the synaptic weights, $\boldsymbol{x}$ is the input vector and $\boldsymbol{\theta}$ is the bias vector.

However, the discrete time representation of DTRNN in many applications is not enough because it is necessary to compute the differential equation of the system. Thus, the Continuous Time RNN (CTRNN) arises to solve the issues related with the discrete time. CTRNN uses a system of ordinary differential equations to model biologically the effects of a spike train into a neuron (Beer, 1995). The operation performed by a CTRNN is shown in Equation 3.8.

$$\begin{gathered}\dot{y}_i(t) = f_i(x_i(t), y_1(t), \ldots, y_n(t)) = \\ \frac{1}{\tau_i} \cdot \left(-y_i(t) + \sum_{j=1}^{n} w_{ij} \cdot \sigma\left(y_j(t) + \theta_j\right) + x_i(t) \right)\end{gathered} \tag{3.8}$$

where, $\dot{y}_i$ is the rate of activation change of postsynaptic neuron, y_i is the activation of postsynaptic neuron, y_j is the activation of the presynaptic neuron, τ_i is the time constant of postsynaptic neuron, w_{ij} is the synaptic weight from pre to postsynaptic neuron, $\sigma(\cdot)$ is the activation function, θ_j is the bias of the presynaptic neuron and x_i is the input to a neuron (if any). Equation 3.8 may be represented in vector notation as:

$$\dot{\boldsymbol{y}}(t) = f_\alpha(\boldsymbol{x}(t), \boldsymbol{y}(t)) = \frac{1}{\boldsymbol{\tau}} \cdot \left(-\boldsymbol{y}(t) + \boldsymbol{W} \cdot \sigma\left(\boldsymbol{y}(t) + \boldsymbol{\theta}\right) + \boldsymbol{x}(t)\right); \tag{3.9}$$

where, $f_\alpha(\cdot) : D \subset \mathbb{R}^2 \longrightarrow \mathbb{C}^2$ is a function dependent of the variables of the system (the synaptic weight matrix $\boldsymbol{W}$ and the bias vector of the neurons $\boldsymbol{\theta}$).

CTRNNs have been applied in different fields including evolutionary robotics, in which they have tackled different problems such as vision, co-operation and minimally cognitive behavior (Beer, 1997). The decision to use discrete or continuous time neural models depends on the application, but continuous time has advantages over its discrete version. One advantage of using continuous time neural models versus discrete ones is that the state derivative is clearly well defined through explicitly with the use of calculus and also facilitates its representation without using time indexes. The problem is that a continuous system could not be simulated directly in a computer due to digital reasons. Thus, the continuous model is transformed in a discrete one where the differential equations are converted to equivalent difference equations which are formally identical representations. However, more sophisticated and faster techniques can be used to solve them rather than difference equations. Another advantage of using continuous time is the possibility of changing the length of the time step to compute the derivative in order to suit changing circumstances without retraining the network. Moreover, continuous neurons tend to retain information better through time in applications temporally continuous (Beer, 1997). An interesting property of the continuous model is the capacity to retain its state through time which is convenient also for non temporal tasks. On the other hand, for discrete neurons, there is no reason that their state at one point in time have a relationship to their state at the next point in time. Maintaining the internal state of the neurons with a slow decay during time, makes possible to speed up the learning process (Pearlmutter, 1990).

CTRNNs have more beneficial features than DTRNNs. Therefore, a CTRNN has been selected among all the ANN structures and architectures for the purposes of this Thesis in order to solve the problem at hand. The reasons to use this type of architecture are based on their dynamical properties, the intensive use of these structures in different fields of application, the continuous temporal representation, the fast adaptation property to the external changes and the minimum knowledge needed from the environment. All these features are desirable to build a system that helps to manage an electrical grid in real time from the demand side through little knowledge of the rest of the nodes that integrate the system. Before continuing, it is necessary to clarify one important issue present in all dynamic systems which is the stability of the RNN which will be addressed in Section 3.3.1.

3.3.1 Stability

Stability is a property of dynamic systems to study the solutions of the differential equations and their trajectories, remaining inside or in the neighborhood of a set of points called orbit. There are different stability criteria to address this property for a system. For example, the Bounded Input Bounded Output (BIBO) stability criterion is normally used to refer to the stability of the input and output of a system. According to this criterion, the stability of a system depends on the growth of the output, which is bounded due to bounded input, initial condition or unwanted disturbance. In general, RNNs are systems that meet the BIBO stability criterion. This criterion is within the structure of the RNNs as they are built with activation functions that present saturation nonlinearities and limit the growth of the output. Thus, this stability criterion does not provide any relevant information to the stability of a RNN. Normally, the stability criterion of nonlinear dynamic systems is related with the Lyapunov stability theory (Lyapunov, 1992).

Before describing the Lyapunov theory, it is necessary to understand some concepts. A dynamic system is a system whose state varies with time and different models can be used to do it. Usually, dynamic systems are represented by a state-space models which consist of a set of state variables whose values possess enough information to predict the temporal evolution of the system (Haykin, 2009). The mathematical expression of this model is described in Equation 3.10.

$$\frac{d}{dt}\boldsymbol{x}(t) = \boldsymbol{F}(\boldsymbol{x}(t)) \tag{3.10}$$

where, $\boldsymbol{x}(t) = [x_1(t), x_2(t), \ldots, x_N(t)]^T$ is the state vector containing the N different states in which the system is divided and $\boldsymbol{F}(\cdot)$ is the nonlinear function evaluated for each state of the system. N is the system order. The function $\boldsymbol{F}(\cdot)$ could depend explicitly on time, autonomous, or not, non autonomous.

Equation 3.10 could be adapted to represent the RNN activations so the dynamic system theory can be applied to them. The state space representation is useful because it provides a visual and conceptual tool to analyze the dynamics of a system rather than numeric solutions of the system. So, the system has been divided in N states that evolve during time. These changes in time can be represented as a curve in the state space, and it is called trajectory or orbit of the system. Equation 3.10 describes the motion of a point in an N-dimensional space, whose solutions are represented in trajectories and the tangent vector of each point represents the velocity at the initial condition. For each possible state, there will be a tangent vector and the assembly of the tangent vectors is described as a vector field. Then, the vector function $\boldsymbol{F}(\boldsymbol{x}(t))$ can be seen as a velocity vector field, or a vector field. The trajectories for different initial conditions can be grouped in the system state portrait, which includes all the system points in which the system is defined and possesses information about the flow of the dynamic system (Guckenheimer and Holmes, 1983).

The solution uniqueness have to be also guaranteed for the state space equation only under certain conditions, which restricts the form of $\boldsymbol{F}(\boldsymbol{x}(t))$. A solution to the system exists if $\boldsymbol{F}(\boldsymbol{x}(t))$ is continuous in all its arguments but it is necessary to meet the Lipschitz condition for uniqueness (Haykin, 2009). Once the solutions to Equation 3.10 are found, the stability of those solutions can be analyzed based on the equilibrium of the states. To begin with the stability study of the states, it is necessary to define when a state is at equilibrium.

Let $\boldsymbol{x}^* \in D$ be an equilibrium state of the system, such that $\boldsymbol{F}(\boldsymbol{x}^*) = 0$. Normally, the equilibrium state is also named singular point because the trajectory can be transformed into a point itself. At the equilibrium state, ${d\boldsymbol{x}(t)}/{dt} \to 0$ and therefore a possible solution to the system is the constant function $\boldsymbol{x}(t) = \boldsymbol{x}^*$. Because of the uniqueness of solution, no other curve will pass through $\boldsymbol{x}^*$. Thus, to study the stability of a system such as Equation 3.10, it is necessary to analyze the behavior of the orbits near to an equilibrium state (Kuznetsov, 1998). The nonlinear function $\boldsymbol{F}(\boldsymbol{x}(t))$ is considered to be smooth enough in order to be linearized in the neighborhood of $\boldsymbol{x}^*$. If the vector function $\boldsymbol{F}(\cdot)$ is a C^1 vector field (the first derivative of each component of the field are continuous), then it can be approximated by its first order Taylor expansion:

$$\begin{aligned} \boldsymbol{F}(\boldsymbol{x}) &= \boldsymbol{F}(\boldsymbol{x}^*) + D\boldsymbol{F}(\boldsymbol{x}^*) \cdot (\boldsymbol{x} - \boldsymbol{x}^*) + r(\boldsymbol{x} - \boldsymbol{x}^*) = \\ &= \boldsymbol{x}^* + D\boldsymbol{F}(\boldsymbol{x}^*) \cdot (\boldsymbol{x} - \boldsymbol{x}^*) + r(\boldsymbol{x} - \boldsymbol{x}^*) \end{aligned} \tag{3.11}$$

where, the rest of terms $r(\boldsymbol{x} - \boldsymbol{x}^*)$ satisfy

$$\lim_{\|\boldsymbol{x}-\boldsymbol{x}^*\|\to 0} \frac{\|r(\boldsymbol{x} - \boldsymbol{x}^*)\|}{\|\boldsymbol{x} - \boldsymbol{x}^*\|} = 0 \tag{3.12}$$

It is expected that the behavior of the solutions of Eq. 3.10 near $\boldsymbol{x}^*$ is qualitatively similar to the solutions of the linearized system.

$$\boldsymbol{x}(t) = \boldsymbol{x}^* + D\boldsymbol{F}(\boldsymbol{x}^*) \cdot (\boldsymbol{x}(t) - \boldsymbol{x}^*) \tag{3.13}$$

where, the Jacobian matrix $D\boldsymbol{F}(\boldsymbol{x}^*)$ is calculated as follows,

$$D\boldsymbol{F}(\boldsymbol{x}^*) = \left.\frac{\partial \boldsymbol{F}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}^*} \tag{3.14}$$

Then, Equation 3.10 at the equilibrium state could be rewritten using Equations 3.11 and 3.13, obtaining

$$\frac{d}{dt}(\boldsymbol{x}(t) - \boldsymbol{x}^*) = D\boldsymbol{F}(\boldsymbol{x}^*) \cdot (\boldsymbol{x}(t) - \boldsymbol{x}^*) \tag{3.15}$$

and as the Jacobian matrix $D\boldsymbol{F}(\boldsymbol{x}^*)$ is nonsingular, i.e. the inverse matrix $(D\boldsymbol{F}(\boldsymbol{x}^*))^{-1}$ exists, then the dynamic system approximation of Equation 3.15 is enough to study the local behavior in the neighborhood of the equilibrium state $\boldsymbol{x}^*$. Thus, the system study is reduced to analyze the Jacobian matrix $D\boldsymbol{F}(\boldsymbol{x}^*)$, specifically it consists of analyzing the eigenvalues of $D\boldsymbol{F}(\boldsymbol{x}^*)$. Then, the behavior of the system will be classified depending on the nature of those eigenvalues, which are calculated as shown in Equation 3.16.

$$D\boldsymbol{F}(\boldsymbol{x}^*) \cdot \boldsymbol{v} = \lambda \cdot \boldsymbol{v}; \ (D\boldsymbol{F}(\boldsymbol{x}^*) - \lambda \boldsymbol{I}) \cdot \boldsymbol{v} = 0 \tag{3.16}$$

where, $\boldsymbol{I}$ is the identity matrix, $\boldsymbol{v}$ is an eigenvector and λ is the eigenvalue. $\boldsymbol{v}$ is the direction in which the linear transformation $D\boldsymbol{F}(\boldsymbol{x}^*)$ is applied, its magnitude only changes by the value of λ. λ characterizes different behavior classes of the system, e.g. for a second order system the eigenvalues and their behavior are described in Table 3.3. In order to stabilize the system, it is necessary that the λ module is smaller than the unit circle ($|\lambda| < 1$) (Kuznetsov, 1998).

Now, this tool allows identifying the local stability of an equilibrium state. Hence, if trajectories tend to be near $\boldsymbol{x}^*$ in a relative amount of time, $\boldsymbol{x}^*$ is stable. On the other hand, if trajectories are repelled over time, $\boldsymbol{x}^*$ is unstable. The method of calculating the eigenvalues is well defined when the dimensionality of the problem is low. However, RNNs are not dynamic systems with low dimensionality, since a RNN could be made by dozens of neurons. Thus, it is necessary to define the stability concepts to understand and apply them to a dynamic system. These definitions are as follows (Strogatz, 1994):

- *Uniform stability.* $\boldsymbol{x}^*$ is uniformly stable if $\forall \epsilon > 0 \, \exists \delta > 0 \mid ||\boldsymbol{x}(0) - \boldsymbol{x}^*|| < \delta \Rightarrow ||\boldsymbol{x}(t) - \boldsymbol{x}^*|| < \epsilon; \ \forall t > 0$. The definition means that a trajectory of the system can stay within a neighborhood of $\boldsymbol{x}^*$ if the initial state $\boldsymbol{x}(0)$ is close to the equilibrium state, otherwise it would be unstable.

- *Convergence.* $\boldsymbol{x}^*$ is convergent if $\exists \delta > 0 \mid ||\boldsymbol{x}(0) - \boldsymbol{x}^*|| < \delta \Rightarrow \lim_{t\to\infty} \boldsymbol{x}(t) = \boldsymbol{x}^*$. Thus, if the initial state $\boldsymbol{x}(0)$ of a trajectory is close to the equilibrium state $\boldsymbol{x}^*$, then the trajectory $\boldsymbol{x}(t)$ will reach $\boldsymbol{x}^*$ when time approaches infinity.

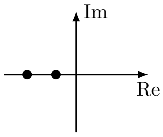
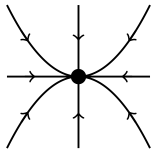
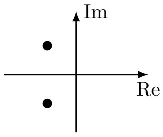
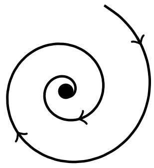
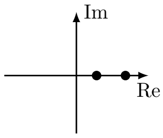
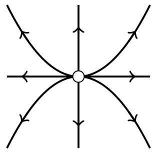
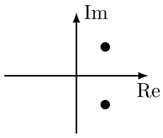
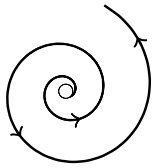
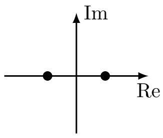
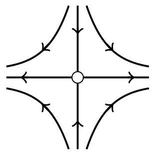
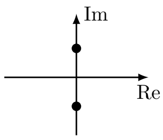
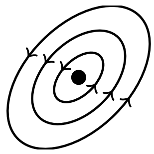
Form	λ	Phase Portrait	Name	Stability
$\lambda_1,\, \lambda_2 \in \mathbb{R}^-$			Node	Stable
$\lambda_1,\, \lambda_2 \in \mathbb{C}$ $\mathrm{Re}\{\lambda_1,\, \lambda_2\} < 0$ $\lambda_1 = \overline{\lambda_2}$			Focus	Stable
$\lambda_1,\, \lambda_2 \in \mathbb{R}^+$			Node	Unstable
$\lambda_1,\, \lambda_2 \in \mathbb{C}$ $\mathrm{Re}\{\lambda_1,\, \lambda_2\} > 0$ $\lambda_1 = \overline{\lambda_2}$			Node	Unstable
$\lambda_1 \in \mathbb{R}^+$ $\lambda_2 \in \mathbb{R}^-$			Saddle	Unstable
$\lambda_1,\, \lambda_2 \in \mathbb{C}$ $\mathrm{Re}\{\lambda_1,\, \lambda_2\} = 0$ $\lambda_1 = \overline{\lambda_2}$			Center	Stable

Table 3.3: Equilibrium states of a second order system.

- *Asymptotic stability.* $\boldsymbol{x}^*$ is asymptotically stable if the equilibrium state is stable and convergent at the same time.

- *Global asymptotic stability.* $\boldsymbol{x}^*$ is globally asymptotically stable if it is stable and all the trajectories of the system converge to the equilibrium state $\boldsymbol{x}^*$ as time tends to infinity. This definition implies that there is no other equilibrium state since all the trajectories are stable and converge to $\boldsymbol{x}^*$. Therefore, the system is always bounded and remains in a steady state for any choice of initial conditions.

These general definitions are difficult to apply in a broad sense to any general dynamic system. In addition, when the dimensions are larger it is not so easy to see if a system has converged to its equilibrium state. So, finding all the solutions to the dynamic system in this way is nearly impossible, that is why a more powerful tool is needed. The solution to find the system stability is found in modern stability theory, in particular in Lyapunov (1992). Normally, the direct method of Lyapunov is applied to find the system stability and consists of using a continuous scalar function of the state which is called Lyapunov function ($V(\boldsymbol{x})$). According to Lyapunov, its function $V(\boldsymbol{x})$ has to be a positive-definite function. Such function must meet that in a neighborhood M of the equilibrium state $\boldsymbol{x}^*$, $\forall \boldsymbol{x} \in M$:

- $V(\boldsymbol{x})$ has continuous partial derivatives with respect to the components of $\boldsymbol{x}$.
- $V(\boldsymbol{x}^*) = 0$.
- $V(\boldsymbol{x}) > 0 \, \forall \boldsymbol{x} \neq \boldsymbol{x}^*$.

Then, Lyapunov used $V(\boldsymbol{x})$ to define Theorem 1 and 2 based on the stability and asymptotically stability for a system of the form of Equation 3.10.

Theorem 1. *The equilibrium state $\boldsymbol{x}^*$ is stable if in a small neighborhood L of $\boldsymbol{x}^*$ there exists a positive definite function $V(\boldsymbol{x})$ whose derivative with respect to time is negative semidefinite in L, i.e. mathematically that*

$$\frac{d}{dt}V(\boldsymbol{x}) \leq 0 \quad \forall \boldsymbol{x} \in L - \boldsymbol{x}^*$$

Theorem 2. *The equilibrium state $\boldsymbol{x}^*$ is asymptotic stable if in a small neighborhood L of $\boldsymbol{x}^*$ there exists a positive definite function $V(\boldsymbol{x})$ whose derivative with respect to time is negative definite in L, i.e. mathematically that*

$$\frac{d}{dt}V(\boldsymbol{x}) < 0 \quad \forall \boldsymbol{x} \in L - \boldsymbol{x}^*$$

Based on Theorems 1 and 2, the global asymptotic stability is defined using the $V(\boldsymbol{x})$ as in Corollary 1.

Corollary 1. *The equilibrium state $\boldsymbol{x}^*$ is global asymptotic stable if in a small neighborhood L of $\boldsymbol{x}^*$ there exists a positive definite function $V(\boldsymbol{x})$ such that it tends to 0 when time tends to infinity, i.e. mathematically that*

$$\lim_{t \to \infty} V(\boldsymbol{x}) = 0$$

These theorems can be applied without solving the state-space equation of the system, which makes the task easier when the system has a large dimensionality. However, there is not a procedure to find the Lyapunov function $V(\boldsymbol{x})$, but the idea is to try with some positive defined functions and see if the system is stable. In some cases, the energy function of the system serves as $V(\boldsymbol{x})$, otherwise it is a matter of trial and error to find one. Not finding a valid function does not prove the instability of the system. The existence of $V(\boldsymbol{x})$ is a sufficient, but not necessary, condition for stability.

The Lyapunov stability theory establishes a general way to study the stability of dynamic systems. However, this theory is wider for the purposes of analyzing the RNN stability, specifically CTRNN, which is the RNN type selected for the purposes of this Thesis. Thus, in Cohen and Grossberg (1983), a general principle is showed to asses the stability of specific type of ANN. The ANN of Cohen and Grossberg (1983) consists of a nonlinear differential equation of the form described in Equation 3.17.

$$\frac{d}{dt}u_j = a_j(u_j)\left[b_j(u_j) - \sum_{i=1}^{N} c_{ji}\varphi_i(u_i)\right], \quad j = 1, \dots, N \tag{3.17}$$

And it admits a Lyapunov function of the form,

$$E = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} c_{ji}\varphi_i(u_i)\varphi_j(u_j) - \sum_{j=1}^{N}\int_0^{u_j} b_j(\lambda)\frac{d\varphi_j(\lambda)}{d\lambda}d\lambda \tag{3.18}$$

For Equation 3.18 to be a valid definition of Lyapunov function, it is necessary that the following ANN conditions are met:

- The synaptic weights are symmetric, $c_{ij} = c_{ji}$.
- The function $a_j(u_j)$ is nonnegative, i.e. $a_j(u_j) \geq 0$.
- The nonlinear activation function $\varphi_j(u_j)$ satisfies the monotonicity condition, i.e. $\varphi_j(u_j) = \dfrac{d\varphi_j(u_j)}{du_j} \geq 0$.

Finally, the Cohen-Grossberg theorem could be stated as follows:

Theorem 3. *Provided the system of nonlinear differential equations of Equation 3.17 which satisfies the condition of symmetry, nonegativity and monotonicity, admits the definition of a Lyapunov function E of the form of Equation 3.18 that satisfies the condition*

$$\frac{dE}{dt} \leq 0$$

then the neural system of Equation 3.17 is Lyapunov stable in the sense of Theorem 1.

Consequently, this theorem provides a powerful tool to build RNNs guaranteeing that they are stable structures. And it is not necessary to solve the entire system of differential equations, or search for a Lyapunov function that meets the stability conditions. In this Thesis, the Cohen-Grossberg theorem is used to build the neural models of CTRNN, following its condition and assuring that the network is Lyapunov stable.

3.4 Artificial Neural Network in action: Applications

In the first part of this Chapter, the basics of ANNs and different structures commonly used have been explained. The ANN application is present in all parts of its design both its structure and its training. ANNs can be applied almost to any application, but with some considerations such as enough data to train the network or a feasible structure to be trained in enough time. In addition, ANNs cannot be used in applications with not enough data because they do not have enough information to extract features from it.

In spite of not being good for arithmetics and precise calculations, ANNs can extract patterns if there are enough data even with noise. They can always give an answer to the input data even when the input is not complete and are well fitted in applications in which they can infer a function or relationship from observations. ANNs are particularly useful in applications where the function they infer from the data is too complex and there are nonlinear relationships among them. But with all these advantages comes a great disadvantage when the complexity of the problem requires a large neural structure because any interpretation can be drawn from the data. The main problem for large structures is the number of free parameters, such as hidden nodes, synaptic weights, learning rates, etc.

At the same time, implementing an application with an ANN has the advantages already described in Section 3.1.1. Thus, when applying ANNs, it is necessary to consider the pros and cons. Recently, an intensive use of ANNs has been done due to the immense data available at any application and the increase power of computation. ANNs are beginning to expand into a variety of fields different from the classic AI ones, such as financial, marketing, industrial or energy, among other fields of interest. In the application of ANNs, there are differences between candidate, developing and already demonstrated applications. In other words, it is necessary to clarify the status of development, implementation and validation of the ANN in each application. A candidate application is an application that it could be solved potentially by an ANN but for the moment there has not been a successful implementation. Developing applications are those in which the problem has been deeply studied and there exists an ANN prototype that has been trained with a simplified version of the problem. Finally, demonstrated applications are the ones in which an ANN structure has already been used to solve a real problem. With respect to the use of ANNs over time, different categories of applications have been discovered, in which ANNs develop an excellent job and are efficient solving problems in these areas. Those areas are as follows (Jain et al., 1996; Basheer and Hajmeer, 2000; Haykin, 2009):

- *Classification.* This is a classic AI problem consisting of identifying which preset categories belong to a new incoming data. The classification is done based on the previous data observations whose categories or classes are known. ANNs are used in this type of tasks due to their input-output mapping abilities. The classification is done normally through a supervised learning algorithm in which an input, represented by a feature vector, is shown to the network and at the same time the class corresponding to the input is used to adjust the free parameters of the network, obtaining the desired result. In addition, there is an unsupervised version of classification known as clustering, consisting of grouping data without any knowledge based on an internal metric and the similarities or dissimilarities between the input patterns. In any case, the ANN in charge of solving the task extracts the features from the data and obtains an internal representation from them to be assigned to one of the classes. The data could be of different nature, e.g. continuous or discrete inputs, inputs from different sources such as different sensors, and the classification could be done in different ways: categories, ordinal, integer-valued or real-valued. Inside this field, there also exist problems related with pattern recognition, decision making or detection, since they consist of mapping inputs into the corresponding outputs. Among the different applications, in which ANNs have been successfully applied in real classification problems, such as character recognition, speech recognition, EEG waveform classification, fault diagnosis, image classification, cancer detection, etc.

- *Data processing.* Many applications can be found within this field, but all of them are related. All of these applications use a starting set of inputs which are manipulated to obtain meaningful information. ANNs are used to infer a relationship between the input and the output, extracting only the valid information and processing it to solve the task assigned. Therefore, the application itself is the one that considers which information is meaningful and which can be discarded. This scope also covers all applications involving signal processing which consists of treating or transferring the information contained within different physical, abstract or symbolic forms denominated signals. Thus, ANNs models manipulate the information of the signal to obtain different goals, taking advantage of the learning and adaptivity abilities of them. For example, ANNs have been used in filtering tasks which consist of modifying the characteristics of the signal itself to obtain a new signal of the characteristics required by the system. Inside filtering tasks, ANN has been used for tasks involving adaptive filtering such as noise reduction or echo cancellation. In addition, ANNs have been used in: validation (ensuring that the data are correct and useful), compression (consisting of dimensionality reduction), analysis (involved in the collection, organization, interpretation and presentation of data), multiplexing information, image processing, encoding information to send it through a canal, speech processing, etc.

- *Function approximation.* It is also known as model building and consists of finding the relationships that exist between the different data available, if any. Suppose that a set of labeled observations of input-output data have been generated by an unknown function. Then, an ANN is used to estimate the unknown function underlying rules relating the inputs to the outputs. Multilayer ANNs are considered universal approximators that can approximate any arbitrary function to any degree of accuracy (Cybenko, 1989). ANNs are also used to compute regression in the data, which is a special case of function approximation. Function approximation with ANNs is applied typically to two different problems: i) no explicit mathematical expression is available, i.e. the data are obtained from experiments or observations, and ii) substituting theoretical models too complex to analyze where the data from the model are used directly. Related with this task, ANN structures are used to predict time series. A time series is a sequence of successive points over a time interval. ANNs are used in time series forecasting to analyze the data in order to extract meaningful statistics and other characteristics and then predict a future value based on the information extracted plus previously observed values. Forecasting is an important application very useful in decision-making in business, science and engineering. For example, typical applications of time series forecasting with ANNs are: stock market prediction, weather forecasting, electricity demand forecasting, etc.

- *Optimization.* This field of application is related with finding solutions to problems in the most efficient way with the less resources possible. Specifically, optimization consists of finding a solution that maximizes or minimizes an objective function that satisfies a set of constraints. A wide variety of problems in mathematics, statistics, engineering, science, medicine and economics can be interpreted as optimization problems. For example, well known optimization problems are the travel salesman problem or the NP-complete problem. ANNs are used in optimization because they are efficient solving complex and nonlinear optimization problems. ANNs have also been used in fitness approximation which is a method consisting of reducing the evaluations of a fitness function to reach a target solution.

- *Association.* This type of applications establishes a set of relationships between the data, which allow identifying some patterns and reproduce them to a given input. Content-addressable memory is a type of association application which consists of a quick search for a given input to obtain the right output previously stored. An advantage of this type of applications is that the content is accessed directly, so that the output will be retrieved correctly even though the input presents some degree of corruption or it is affected by noise. An ANN for association tasks is trained on noise-free data and then used to classify noise corrupted data as in novelty detection. The network will also be used to correct or reconstruct the corrupted data or completely missing data. For example, Hopfield networks and multilayer perceptrons are used in this type of tasks. In addition, association problems are also related with establishing similarities between states that are close in time or space. Inspired in the characteristic of the brain to organize information, the SOMs are other ANN structures that solve tasks related with association. SOMs are spatial maps of the input patterns, in which the information contained in the input patterns are distributed spatially in the lattice that forms the neurons and the activation of each of them is indicative of intrinsic statistical features of the inputs.

- *Control.* It is a discipline which applies control theory to make a system behave in a wanted or desired way. For example, a dynamic model defined by $\{u(t), y(t)\}$, where $u(t)$ is the control input and $y(t)$ is the resulting output of the system at time t. The goal is to generate a control input $u(t)$ such that the system follows a desired trajectory determined by a reference model. Examples of these implementations are the speed control of a motor or fix the motor to a determined position. Thus, the idea is to design an ANN with the sensory input of the system and elaborate an output that the system will follow based on system feedback. A neural controller is not an easy task to train but with the adequate inputs and the right system design, a nonlinear controller could be used to fulfill the task that otherwise would be too complex. A field related with control is robotics, in which robots are built using control, sensory feedback and information processing. ANNs are used to establish some nonlinear relationships between the sensory inputs of the robot and the desired response of the robot, e.g. manipulators, speech recognition, gestures and artificial emotions. Another control field in which ANNs are being applied is in the prosthesis, which consists of replacing a missing body part with an artificial device. ANNs are used to process the information coming from the muscle around the missing body part and elaborate a response to move the prosthetic in the right way.

In all these applications, the idea of processing information with ANN is present. They also take advantage of their characteristics: i) input-output mapping, ii) nonlinearity and iii) feature extraction. All the applications, involving these characteristics can be solved by an ANN approximation. However, some neural architectures are best suited to solve a certain type of problems, such as SOMs which implement content-addressable memory applications or RNNs which are best for prediction, time-series representation, adaptive filtering, control dynamics, etc. due to the presence of feedback. In this Section, some applications were shown in which ANNs can be applied, but they can also be applied to: security (face identification, biometrics), data mining (or knowledge discovery in databases), medical diagnosis (cancer detection, processing medical images), financial applications (automated trading systems) or system identification (e-mail spam filtering).

There are many fields in which ANNs have been successfully applied but their expansion continues. A promising field of application for ANNs is the electricity field because with the arrival of the Smart Grid (SG) is increasing the need for fast information processing and using their features for prediction and time series representation would help with the management of the grid. As part of this Thesis, the electricity field is strongly presented so Section 3.4.1 reviewed how ANNs have been used in power systems.

3.4.1 Artificial Neural Network in Electrical Applications

With the advent of SG, algorithms are becoming increasingly necessary to help to automate the continuous operation of the grid as its complexity is also increasing. Signal processing is becoming an essential tool to understand, plan, design and operate the complex processes of the grid. Many electrical applications involve the use of signal processing as a tool to analyze them and the need of these techniques is becoming a real fact. The reasons are the vast amount of data available for correlation, diagnosis and analysis inside power systems nowadays (Ribeiro et al., 2014).

There exist several reasons why applying ANNs to power systems, such as their learning ability from the environment, fault tolerance, distributed system, real time application, etc. They are also self-organized structures, which create their own data representation during the learning process in real-time operation, since the computation can be done in parallel and the redundancy of the structure prevents the degradation performance of the network in any fault case (Momoh, 2012). In addition, ANNs need fewer constraints than other algorithms when using them in power system applications. The data used are straight forwardly handled without a significant reduction or manipulation of the variables. Another advantage of such algorithms is that they add value to the investments made in infrastructure within the SG. The reason is that once trained for the task they are undertaking, ANNs can be integrated into new ICT systems without assuming higher costs. Hence, such methods of system performance evaluation are very much cost-effective and will not hinder the consumer welfare by an unnecessary increase in electricity price (Sen et al., 2015). ANNs have been successfully applied in different applications related with power systems: i) security assessment, ii) fault detection and diagnosis, iii) load forecasting, iv) transient stability, v) control analysis, vi) economic dispatch and vii) system protection and design (Haque and Kashtiban, 2007). Some of the most important applications are explained as follows:

- **Load forecasting**. One of the main applications in which ANNs have been applied is load forecasting. The grid operation involves planning the different generators to accomplish the real-time matching of the demand. In addition, load forecasting does not only serve for the system daily operation, but to prevent future failures or system congestion, ensuring that users always receive their electricity. Thus, load forecasting is related to financial, development, expansion and planning, and it is important to make accurate predictions for system enhancement. There exist three different forms to predict the grid demand based on the forecast time length: short-term, mid-term and long-term load forecast.

– *Short-term load forecasting* consists of elaborating a forecast using a temporal interval from an hour to a week which is important for different applications such as real time control, economic dispatch, unit commitment, etc. (Haque and Kashtiban, 2007). For example, Park et al. (1991) used a perceptron to make load forecasts in intervals of hours and days by using the weather and load information. They showed better performance of load forecasting with ANNs than using conventional techniques. In Germond et al. (1993), a SOM was used to short-term forecasting of peak electrical loads by using the load data of a real system. In Hippert et al. (2001), there is a description of different ANN models used for short load forecasting. Different ANN structures have been used, such as multilayer perceptrons, SOM, RNN, RBF, etc., using different input data, like weather conditions, holidays, weekends and special sport matches days inside the forecasting model. But typically, the data used for short term load forecasting are time factors and weather data. In these models different prediction horizons have been used such as 1 hour, 24 hour (a day), 168 hour (a week) for predicting the entire demand curve or peak hours. ANNs in short-term load forecasting are known to solve the limitations of other traditional models. They have overcome the difficulties to find relationships between all variable attributes and instantaneous load demand in order to adapt rapidly to the nonlinear system-load changes.

– *Mid-term load forecasting* covers a prediction horizon time from weeks to a couple of years. The data normally used to elaborate the predictions are different from the ones used to short-term load forecasting. Mid-term forecasting is sensitive to growth factors, which are factors that influence demand such as seasonal variations, addition of new loads, maintenance requirements of large consumers, etc. However, mid-term forecasting is not as accurate as short-term forecasting on power system operations. In Feilat and Bouzguenda (2011), a multilayer perceptron is used to forecast the monthly peak load based on the historical monthly load data, temperature, humidity and wind speed. Normally, the data used to elaborate the forecasts consist of historical load, weather, economics and demographic data (Feilat and Bouzguenda, 2011). With this information, it is possible to plan the maintenance of a certain plant during a time period, major tests in the system, commissioning events, determine outage times of plants, etc. Tsekouras et al. (2006) use an adaptive ANN, which properly transforms the input variables to differences or relative differences, in order to predict energy values not included in the training set. Another example of yearly mid-term load forecasting is described in Bunnoon (2011). An ANN is used to forecast a year ahead the grid demand based on different factors, such as temperature, humidity, wind speed, rainfall, industrial index and consumer price index. It is used for a unit commitment and a fuel reserve planning in the power system.

– *Long-term load forecasting* has the larger prediction horizon and covers a time span from a couple of years to decades. In this type of forecasting, quick changes in the demand are not as significant as in other horizon prediction. Grid operators need accurate forecasts that allow the power system to implement new strategies and continue its expansion. Long-term forecasting as mid-term forecasting uses growth factors to elaborate predictions and take into account long historical data of the power system and other macro economic variables related with power systems, such as demographic growing, time factors, facilities investment or sales, regional development, energy supply price, etc. They are normally used to supply the electric utility companies with information to make investments and take decisions regarding planning (equipment purchases), maintenance (staff hiring) and expansion. An example of a long-term load forecasting is described in (Daneshi et al., 2008). An ANN together with fuzzy logic elaborate their prediction in a volatile electricity market based on the forecast growth of population, monthly temperature of the previous year and previous monthly peak load. Another predictor is described in Achanta (2012), which consists of using a multilayer perceptron and SVM to predict the demand. They use historical load data to do the predictions, but preprocessed to obtain a better result. A last example of long-term load forecasting using RNNs is presented in Hayashi and Iwamoto (1994).

- **System security**. Another application of power system in which ANNs have been used is in security. For example, Chawla et al. (2005) used different feedforward ANN structures to help with the protection of different grid parts. Another security concern is security assessment, which consists of monitoring the power system status guaranteeing that its functions are carried out normally. In the case of power systems, security of supply has to be guaranteed for all the users without exceeding acceptable voltage and/or frequency limits. Thus, security assessment is defined as a function that predicts the vulnerability of the system to possible events on a real-time basis. There are two types of security assessments: static, corresponding to those states where the transients following a disturbance have finished, or dynamic, which corresponds to search for disturbances that may lead to transient instabilities. However, both are responsible for differencing between 3 states: secure state, in which there is no problem inside the system; alert state, in which the power system is reaching is limit capacity; and emergency state, the system is off-limits and the operation is insecure (Swarup and Corthis, 2006). In the literature, many examples of using ANNs for security assessment are found because ANNs fit perfectly to some requisites such as the prediction of failures, real time operation and low economic investment. In Sobajic and Pao (1989), a pattern recognition task implemented with ANNs is done for dynamic security assessment. Whereas in Swarup and Corthis (2006), a SOM is used for the static security assessment of a power system. Another implementation of RBF ANNs is presented in Srilatha et al. (2014).

- **Transient stability**. The grid is a large system prone to different faults, which can be predictable or unpredictable, due to internal (e.g. random load) or external disturbances (e.g. lightning). One of these problems is transient stability, which is related with the loss of synchrony on part of the system due to a large disturbance that can cause in the worst case scenario instability to the whole system. This stability problem is related with oscillations present in the machinery inside power systems, specifically with generators. Electromechanical oscillations are caused by the instantaneous imbalance between generation and consumption and are represented by the exchange of energy among the generator rotors with the interconnected network (Karami, 2011). The transient stability problem is a dynamic nonlinear problem in which ANNs fit perfectly due to their parallel processing abilities and the nonlinear featuring characteristic of their modular structure. For example, in Ostojic and Heydt (1991), an ANN structure is used to achieve transient stability through a pattern recognition methodology in the frequency domain. Another example is the one described in Karami (2011), that uses a multilayer perceptron to estimate the normalized power system transient stability margin through its mapping with the conditions of the power system. In Haidar et al. (2011), a general regression neural network is used for predicting the status of the power system and making a classification for transient stability evaluation in power systems.

- **Fault diagnosis**. One of the major problems for system outages is the failure of equipment, but reliability and security can be improved with the use of proper systems for detection and diagnosis. Therefore, the first step consists of identifying the fault, its nature and location, because when a fault occurs, the system operators have to minimize the impact of the failure and restore completely the system as soon as possible. However, the number of alarms that triggers when a fault occurs makes impossible to detect the source of the failure necessary to restore the system to a secure state. Despite having defined a hierarchy for failure identification, the elapsed time between the start of the problem and its detection is often paramount and it is necessary to act effectively, finding the fault as soon as possible. ANNs have been successfully applied to classify the different failures in apparatus of the power systems such as transformers or switches (Haque and Kashtiban, 2007). The advantages of using ANNs are that the diagnosis is carried out without interrupting the service, their flexibility with noisy data and their classification ability to distinguish between failures. In the literature, many examples of ANN structures are applied to different detection problems. In Ebron et al. (1990), a multilayer perceptron was applied to detect incipient faults in distribution power lines, specifically in the detection of high-impedance faults. It used preprocessed data of the current that goes inside the lines and simulation results showed agreement in most of the cases. Another example is the one described in El-Fergany et al. (2001), which consists of a hybrid system of ANN and expert system for off-line fault diagnosis in power systems. This system uses the information of the operated relays and tripped circuit breakers after they reached their final status. The system proposed is able to indicate in which section the fault has occurred. Finally, in Wang (2011), different approaches using ANNs to real time fault detection and diagnosis are explained. In this case, ANNs are used as pattern recognition for classifying different faults in the power system.

- **Economic dispatch**. The last major application of ANNs in power systems is economic dispatch. This application consists of minimizing the operating costs by optimally meeting the electricity generation to the system load subject to transmission and operational constraints. To achieve the minimum total cost, the generators with the lowest marginal costs must be used, then the next ones until the load is met. In practice, the whole units operating range is not always available for load allocation due to physical operation limitations (Park et al., 1993). In this case, the ANN is used to optimize the resources available and minimize the costs. Traditionally, this problem was solved approximating the cost of each generator as a quadratic function or more accurately as a segmented piecewise quadratic function for plants with more than one type of fuel. There exist different ANN approximations to solve this problem. For example, in Park et al. (1993), a Hopfield network is used to solve this problem by using different segmented piecewise quadratic function to represent the costs of the generation. Other works, such as the one in Yalcinoz and Short (1998), used also a Hopfield network with a mapping quadratic technique to solve economic dispatch with transmission capacity constraints. Another example, using multilayer perceptrons to solve these problems is the one described in Imen et al. (2013). ANNs, specially the Hopfield model, have a well-demonstrated capability of solving combinatorial optimization problems. Because of their capabilities to take into account different power system limitations such as transmission line loss, penalty factor, control pollution of the units, etc., it has experienced an increasing use in this field.

ANNs have been extensively used in power systems, and the variety of applications make difficult to decide which ANN is the best for each task. Only the most important applications are shown in which ANNs have been used, but there exist other uses of ANNs in power systems such as real time monitoring, operation state estimation, modeling power systems, voltage control, reactive power dispatch, etc. (Haque and Kashtiban, 2007). In addition, more applications of ANNs related with the energy field can be found, specifically with the renewable energy applications. A complete review for different applications related with the renewable energy and other energy systems are described in Thiaw et al. (2014). Some of these applications are: modeling a solar steam generator, prediction of solar resource, wind speed prediction, peak power tracking for Photovoltaics (PV) system, load forecasting, tariff forecasting, etc. Thiaw et al. (2014) described the use of ANNs related with two applications of renewable energy: the first one is related with PV systems and consists of tracking the maximum power point of PV generators to extract the maximum power available; and the other application is related with the assessment of a wind energy resource. Both cases used a multilayer perceptron and the electrical parameters of the generator were used as input to the neural model. In the case of the wind assessment, the neural model is used to calculate the wind speed distribution law based on the wind speed.

ANNs are nowadays of great value in different grid applications. Moreover, they are going to represent a great asset in the future development of the SG in which they can be crucial to solve some problems of its implementation. ANNs could be used to cover the following tasks inside the SG:

- *Forecasting*. They could help in the forecasting of renewable sources, integrating them in the power system minimizing energy losses, storage forecasting, favoring the electricity management, or in demand forecasting.

- *Pricing*. Establishing an electricity tariff in real time is very difficult because it depends on many external factors, such as the cost of energy, the time of the day, etc. Therefore, the nonlinear dynamic abilities of ANNs can be used to forecast and adjust a real time tariff of electricity.

- *Demand Side Management (DSM)*. The grid enhancement requires that demand technologies will react dynamically to the status of the grid, not only to control or price signals. The SG requires DSM approaches that are able to adapt to variable situations and learn from the environment in order to make a more efficient use of the supply. ANN controllers could help to control the demand and be dynamic responsive to a changing environment.

- *Data processing*. With the arrival of the smart meters, the grid operators can have information in real time from each part of the grid, such as real-time prices, peak loads, network status, etc. Hence, ANNs can be used to manage this vast amount of information usefully, inferring some relationships and conclusions from them to support the decision making process.

- *Intelligent diagnosis*. ANNs are a great option to fault diagnosis because they are able to make nonlinear relationships between the data coming from the grid being able to make better diagnosis. They are also fault tolerant, that can handle corrupt data because they are ready to interpolate data and learn from them to elaborate a more accurate response. And finally, their mapping abilities make them appropriate to extract relationships between input and output in fault detection and diagnosis applications.

- *Protection*. Some of the above examples show that they are suitable for fault detection and they could be perfect for detecting problems in microgrids, being able to isolate them from the grid in case of fault.

- *Security*. There are many concerns about the security in the SG, because of the vulnerability of the communications. It can be exposed to cyber attacks that can interfere with the structure of the system. However, ANNs are robust enough to withstand these attacks and in spite of part of the network going down, the rest could do the task at hands. They can also serve to encode the information, sending it securely avoiding eavesdropping.

ANN can play different and important roles inside many application fields of power systems. They are well fitted to develop almost any task inside the grid, when there is enough data and time to train them to handle the task. But how can an ANN be trained to do a task? Section 3.5 describes different methods to train ANNs.

3.5 How to train your ANN: Learning vs. Tuning

ANNs need to be trained in order to obtain the adequate response for the environment stimulus and solve the task at hands. The first way that someone can come up with the weight training of an ANN would be their manual adjust. However, this method is not effective for large ANNs of more than two or three neurons because there are too many parameters to be adjusted. Thus, two different forms to train an ANN are introduced in this Section. The first one is based on traditional learning algorithms in which the ANNs learn from their environment and presumably is most closely to the biology interpretation of the training. Meanwhile, the second one is based on optimization techniques, specifically Genetic Algorithm (GA) which consists of adjusting the free parameters of the network based on a cost function to minimize its error.

3.5.1 Learning

A precise general definition of learning is difficult to enunciate. However, a learning process in ANN sense can be defined as the problem to update the network architecture (w_{ij} and θ_i). Thus, it can perform the task for which it was built, producing a desired output for a given input (Haykin, 2009). The basic idea behind a learning system is that it changes itself to adapt dynamically to external conditions such as the environment surrounding it. Once the network is trained, it can be used to perform the task and it can respond even for situations in which it has never been trained due to its generalization property. An ANN acquires its knowledge from the environment through the available training samples or patterns.

But how is an ANN able to learn or be trained with external data? First of all, it is necessary to think which part or parts of the network are available to be modified in order to learn a task. Remembering Section 3.1.1, an ANN possesses synaptic weights, bias, propagation rules, activation functions and output functions to be configured. Thus, an ANN could learn from various modifications of its parts, such as i) developing new connections, ii) deleting existing ones, iii) altering the value of the synaptic weights among neurons, iv) changing the neuron bias, v) varying the internal elements of the neural functions (propagation, activation and/or output functions), vi) adding new neurons and vii) removing neurons. These are all the possibilities available to affect the behavior of the network, some of them are more plausible than others.

Some possible modifications described are not usually done due to its complexity and interpretation. For example, modifications of the functions and their parameters are rarely used, because in the development of a learning procedure is difficult to create relationships between the data and their modifications. Moreover, it is not biologically plausible and is not intuitive how changes affect the realization of the task at hands. Therefore, the creation (neurogenesis) or destruction (apoptosis) of neurons is also a not common learning modification, although it can provide well adjusted weights during the training and also an optimized network topology. These modifications are also biologically plausible and are growing interest but they are usually implemented with other techniques, not with learning algorithms. The reason is that learning algorithms traditionally are focused on acquiring the knowledge from the environment for a given structure.

Therefore, the more common rule used to learn from the available data is the modification of the synaptic weights and bias. Changes in the value of w_{ij} are related with the biological plasticity of the nervous system and the existing connections of the neurons. When a connection ceases to exist, the value of the $w_{ij} = 0$ and when a connection is created, w_{ij} has a value different from 0. Modifications of θ_i are related with the neural stimulation and its activation, so that for negative values the neuron is inhibitory and for positive ones is excitatory. In order to simplify the implementation of the learning algorithms, θ_i is treated as a w_{ij}. Thus, this is how the ANN learns from the environment and stores all the knowledge in its structure through its modification from a learning process. The acquired knowledge is then used by the ANN to interpret, predict and respond appropriately to the environment (Haykin, 2009).

These modifications are carried out based on the external information coming from the environment and a set of rules used to build the algorithms. Then the performance of the network is improved by iteratively updating the matrix of connections ($\boldsymbol{W}$, containing also θ_i). And finally, the ANN learns input-output relationships from the given collection of representative examples. These examples

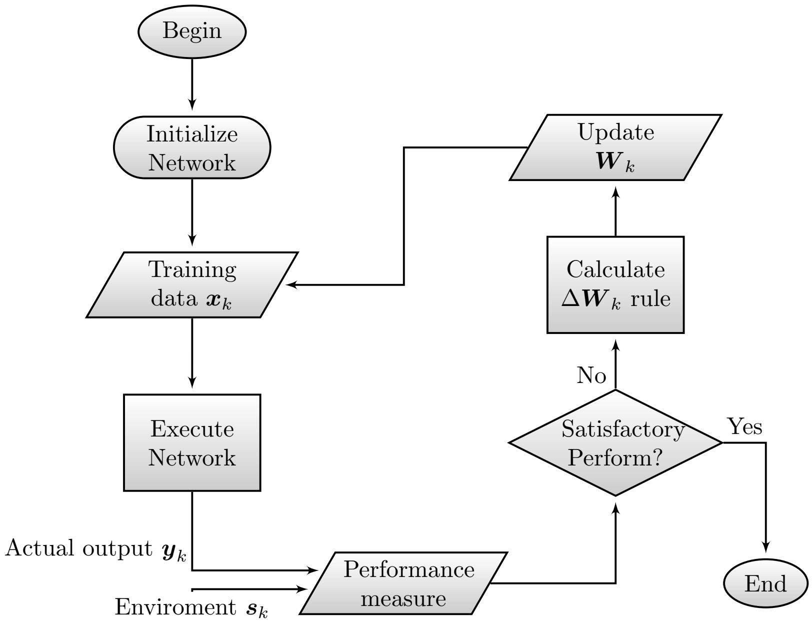


Figure 3.8: Flow diagram of a general purpose learning algorithm.

must possess significant information, so that the ANN can extract the appropriate relationships. The information coming from these data examples could be related with the environment, such as its state, facts about what is, or observations, which consist of measurements from the environment through sensors. In order to design a learning process, the environment in which the ANN operates must be known, i.e. what information is available to the network.

Figure 3.8 shows the corresponding flow diagram of a learning algorithm. It can be observed the different parts in which a learning process is divided, been relatively easy to be implemented by means of a programming language. The example of Figure 3.8 about a general purpose learning algorithm is as follows:

- The first step consists of initializing the ANN parameters, functions, w_{ij} and θ_i and all the elements needed by the learning algorithm to perform its task, environment information, learning rule, etc.

- Then, the next step consists of selecting one of the input patterns ($\boldsymbol{x}_k$) available in the training dataset consisting of a set of inputs from the information available in the environment.

- After this, the network is executed with its actual structure without being modified and its actual output or performance ($\boldsymbol{y}_k$) is obtained.

- In the next measurement, the performance of the network is evaluated to obtain an objective of how well or bad the network is performing its assigned task. Depending on the learning paradigm, an external signal coming from the environment (s_k) is used to elaborate this measure. In some cases, s_k could be the desired output, in others could be only the information about how well or bad the network is doing the task and there are cases in which this signal $s_k = 0$. s_k is integrated inside the training dataset, for each step of the training.

- Then, a stopping criterion is contrasted with the performance of the network. This criterion could be based in reaching a determined value of the objective measurement selected, such as an error, or a historical value of this measure, reach the number of cycles to train the networks, etc:

- If the network exceeds this criterion, then the learning process ends. Otherwise, the learning process continues and it is necessary that the network assimilates the result of the objective measurement into its structure.
- In case that the stopping criteria is not satisfied, it is necessary to calculate the modification of the free parameters of the network, $\Delta \boldsymbol{W}_k$, taking into account the measurements done previously.

- Finally, the structure of the network is modified by following $\Delta \boldsymbol{W}_k$ and a new step of the training begins by taking the next example inside the training data set.

Those are the common steps in a general purpose learning algorithm and the name of a complete cycle is called "epoch". There are different existing learning algorithms, but all of them follow a structure similar to the one of Figure 3.8. However, not all of them apply these steps in the same order, or the information coming from the environment is different or simply use a different variable to update and change the behavior of the network. Thus, there exists a vast amount of different learning algorithms. In any case, the versatility provided by learning algorithms makes possible that ANNs perform almost any task whenever there are enough training data and enough time to train them. In Sections 3.5.1.1 and 3.5.1.2, different types of training and some concerns to consider when training an ANN are presented.

3.5.1.1 Types and rules

This Section introduces different types of learning algorithms and the rules they use to modify the different free parameters. In Section 3.2, when classifying the different neural architectures, three different paradigms of learning were introduced: supervised, unsupervised and reinforcement. The difference between them is how the algorithm interacts with the environment and how much information is obtained from it. Therefore, from Section 3.2, the definition of each learning paradigm is (Jain et al., 1996; Haykin, 2009):

- *Supervised.* It is also called learning from a teacher. In this type of learning, the ANN is provided with information about the desired output of the system for each input. The modification of the weights is based on the deviation of the ANN output from the desired output for the given input.

- *Unsupervised.* There is no environmental signal s_k coming from the outside, the only information available is the set of inputs. This learning algorithm explores the structure in the data to detect similarities among them, doing correlations between patterns in the data, and organizing them into categories from these correlations.

- *Reinforcement.* As in supervised learning, there is information coming from the environment about the performance of the network. But rather than using the desired output, the network is provided with only an observation on the correctness of ANN outputs or possibly, how right or wrong the ANN output was. Thus, there is not an exact measure of the error, only information about the performance of the network.

Each of these algorithms can be used alone or combinations of them using *hybrid* learning algorithms, in which different learning strategies can be used to train different parts of the ANN. The more biologically plausible of all these learning paradigms is the unsupervised learning, because only the input is given to the network and it is the ANN which establishes the relationships among the data. However, this paradigm is not suitable for all the problems. In contrast, reinforcement and supervised learning have information from the environment, consisting of how well the ANN is doing its task, but in reinforcement it is less precise than in supervised learning. Supervised learning algorithms are the most extended ones because it is easy to establish different relationships between the data when both input and output information are available. With this type of learning paradigm, it is perfectly characterized how the ANN has to behave, but in some tasks the information might not be available, requiring the use of other learning paradigms.

Another classification can be made based on the update rate of the $\boldsymbol{W}_k$. This classification is related with the learning cycle length or epoch and it is divided in offline and online learning. Thus, *offline learning* is referred to learning algorithms which consist of modifying the ANN parameters after several input dataset entered at once, and then, the cumulative objective measurement is used to make the update of these parameters. While *online learning* consists of updating the ANN each step after every input pattern is executed. Offline learning is also called batch learning, because the epochs are organized in data batches. The benefits of using offline or online depend on the problem to solve with them. Online learning algorithms are more suitable for changing and continuous data from the environment since each step they are evaluating the ANN performance. On the other hand, offline is more suitable to steady environments in which the data are not changing so quickly and more complex calculations could be done to extract the ANN performance. Online is simpler than offline due to the fact that it only processes one data at a time. Offline needs that the data are shuffled, so that in each input batch the information of every input type is represented. Thus, online learning is a more general framework than offline (Basheer and Hajmeer, 2000).

Finally, the learning algorithms can also be classified by the rules that they use to update the free ANN parameters. There exist four types of learning rules, which are (Jain et al., 1996; Basheer and Hajmeer, 2000; Wilamowski, 2009):

- **Error-correction rules**. These rules are used in supervised learning paradigms. The ANN performance is evaluated during the training with the arithmetic difference (error, $\boldsymbol{e}_k$) between the actual output ($\boldsymbol{y}_k$) and the desired output ($\boldsymbol{d}_k$) given to the actual epoch. Thus, $\boldsymbol{e}_k = \boldsymbol{d}_k - \boldsymbol{y}_k$ is used to modify the parameters of the network by minimizing the error function overall training samples. Some of the most frequent error correction rules are as follows:

  – *Perceptron learning rule.* Developed by Rosenblatt to train a simple perceptron during a classification task, this rule established a boundary to separate the data in different types. Rossenblatt proved in the perceptron convergence theorem that when training patterns are drawn from two linearly separable classes, the perceptron learning procedure converges after a finite number of iterations (Rosenblatt, 1958). Equation 3.19 shows how the update of the ANN parameters is computed.

  $$\Delta \boldsymbol{W}_k = \eta \cdot (\boldsymbol{d}_k - \boldsymbol{y}_k) \cdot \boldsymbol{x}_k \tag{3.19}$$

  where, η is the learning rate at which $\boldsymbol{W}_k$ is modified.

  – *Correlation learning rule.* This learning rule is similar to the Hebbian learning rule but in supervised learning algorithms. The idea is that the connections between neurons that fire simultaneously have to be positive while the ones between neurons with opposite reactions should be negative. This means that the weights should be proportional to the product of the learning rates. Mathematically, this learning rule is as follows:

  $$\Delta \boldsymbol{W}_k = \eta \cdot \boldsymbol{d}_k \cdot \boldsymbol{x}_k \tag{3.20}$$

  Usually, $\boldsymbol{W}_k$ is initialized to zero to start from an equilibrium state.

  – *Outstar learning rule.* This rule is applied in networks where the inputs and $\boldsymbol{W}_k$ are normalized and it was developed in Grossberg (1969). Thus, the connections between nodes should be equal to the desired output $\boldsymbol{d}_i$ of that neuron.

  $$\Delta \boldsymbol{W}_k = \eta \cdot (\boldsymbol{D}_k - \boldsymbol{W}_k) \tag{3.21}$$

where, $\boldsymbol{D}_k$ is the matrix of the desired outputs for each neuron of the network and η is a constant learning rate of small value that decreases during the training when the algorithm converges to the solution.

– *Widrow-Hoff LMS learning rule.* Widrow and Hoff (1960) developed a supervised learning algorithm that consists of minimizing the error of the output of each node (y_j) with respect to the desired output (d_j) during a batch training of P input patterns:

$$\begin{gathered} e_j = \sum_{p=1}^{P} (d_{jp} - y_{jp})^2 \\ \textit{with } \; y_j = \sum_{i=1}^{N} w_{ij} \cdot x_i \end{gathered} \tag{3.22}$$

where, e_j is the error for the jth neuron, y_{ij} is the output of the jth neuron for the ith pattern and d_{ij} is the desired output of the jth neuron for the ith pattern. Then, the update of $\boldsymbol{W}_k$ is based on minimizing Equation 3.22 by calculating the gradient of the error with respect to the weights:

$$\begin{gathered} \Delta w_{ij} = -\frac{1}{2} \cdot \eta \cdot \frac{\partial e_j}{\partial w_{ij}} \\ \textit{with } \; \frac{\partial e_j}{\partial w_{ij}} = -2 \cdot x_i \sum_{p=1}^{P} (d_{jp} - y_{jp}) \end{gathered} \tag{3.23}$$

This rule is also known as LMS rule and the weights are updated following the gradient descent of the error. The weights are corrected after a number of input patterns presented to the network so it is a cumulative rule. This rule usually leads to a faster solution, but it is sensitive to the order in which patterns are applied.

– *Linear regression.* This rule is similar to the LMS, but it only works with linear neurons and the operation of each neuron is equal to the desired output, i.e. $\boldsymbol{xW} = \boldsymbol{d}$. Thus, solving this matrix equation, the weight values needed to obtain the desired output are found.

– *Delta learning rule.* This algorithm is the generalization of the LMS rule also developed in Widrow and Hoff (1960), and the difference is in the neuron used, which is not linear. Hence, the error of Equation 3.22 has the same form but in this case, it is equal to

$$y_j = \sigma_j \left(\sum_{i=1}^{N} w_{ij} \cdot x_i \right) = \sigma_j \left(\nu \right)$$

Then, the gradient is computed again and the result is in Equation 3.24.

$$\frac{\partial e_j}{\partial w_{ij}} = -2 \cdot \sum_{p=1}^{P} (d_{jp} - y_{jp}) \cdot \frac{\partial \sigma_{jp}(\nu)}{\partial \nu} \cdot x_i = 2 \cdot x_i \cdot \sum_{p=1}^{P} \sigma'_{jp}(\nu) \cdot (d_{jp} - y_{jp}) \tag{3.24}$$

Finally, the modification of the weights is calculated as,

$$\Delta w_{ij} = -\frac{1}{2} \cdot \eta \cdot \frac{\partial e_j}{\partial w_{ij}} = \eta \cdot x_i \cdot \sum_{p=1}^{P} \delta_{pj} \tag{3.25}$$

The weight change is proportional to the input signal x_i, to the difference between desired and actual outputs $(d_{jp} - y_{jp})$, and to the derivative of the activation function $(\sigma'_{jp}(\nu))$. The weights can be updated by using incremental and cumulative methods depending on the number of training patterns used until the new update. This is a useful training rule for only a single layer network, problems arise when there are more than one or when there is no desired output for all the neurons of the network.

– *Backpropagation* This is the most famous learning algorithm for supervised learning algorithms (Rumelhart and McClelland, 1986). It has been applied to multiple applications and is extended because it allows training a network in less time than other algorithms and with not too much computation. The idea of the backpropagation consists of applying the delta rule to each of the different neurons but in a particular way. It can be divided in two phases: propagation and weight update. Propagation consists of executing the network forward with a training pattern to produce the output of the network. Once the outputs of each neuron are obtained, it is necessary to calculate the delta rule of each one or the error gradient. For the output layer, the delta rule of Equation 3.24 will be applied directly because there is a desired output. However, for hidden or input neurons, it is necessary to calculate it with respect to the global error. The value of the gradient for a hidden neuron is calculated as:

$$\delta_k = -\frac{\partial e_j}{\partial \nu_k} = -\sigma'_k(\nu_k) \sum_j \delta_j \cdot w_{jk} \tag{3.26}$$

where, $\sigma'_k(\nu_k)$ is the derivate of the activation function with respect to the propagation rule, δ_j is the local gradient or the delta rule of the next layer and w_{jk} are the connections between the hidden neurons and the neurons of the next layer. Thus, the algorithm consists of obtaining all the information necessary to calculate the local gradient of each neuron and then, it applies Equation 3.27 in order to update each weight.

$$\Delta w_{ij} = \eta \cdot x_i \cdot \sum_{n=1}^{N} \delta_n \tag{3.27}$$

So neurons in the output layer are only modified by their own performance, but for neurons in previous layers, the connection to each neuron of the following layer will be influenced to change its value. Thus, this is one of the problems of this learning rule because when the number of layers is too large, the error value transmitted from the output layer to the input one decreases as it passes through more layers.

- **Boltzman learning**. This type of learning is a stochastic learning rule inspired by information-theoretic and thermodynamic principles. It is used in Boltzman machines which are RNNs with binary neurons. These neurons are stochastic units that generate a state according to the Boltzmann distribution of statistical mechanics and are divided in two subsets, one visible and one hidden. It operates in two modes: clamped, in which only visible neurons operate, and free-running, in which all the neurons can operate freely. So, the objective of the Boltzman learning is to adjust the weights in order that the performance of the visible neurons of the network satisfies a particular desired probability distribution. The Boltzman learning rule consists of

$$\Delta w_{ij} = \eta \cdot (\bar{\rho}_{ij} - \rho_{ij}) \tag{3.28}$$

where, η is the learning rate, and $\bar{\rho}_{ij}$ and ρ_{ij} are the correlations between the states of the neurons i and j when the network operates in each mode. Boltzman learning is a special case of error-correction learning but rather than using an error measure, it uses the difference of the correlation between the outputs in each mode of operation.

- **Hebbian rule**. This is the first learning rule used to train an ANN and it was used by Hebb (1949) from the observation of neurobiological experiments. This rule is based on the assumption that if the synapse of two neurons that are activated synchronously and repeatedly, this synapse should increase its value. While for neurons that operate in the opposite phase, the value of the weight should be decreased. Equation 3.29 includes these assumptions.

$$\Delta w_{ij} = \eta \cdot x_i \cdot y_j \tag{3.29}$$

where, x_i and y_j are the output values of neurons i and j, which are connected with the corresponding synaptic weight. In this case there is not any information about the desired output of the neurons, so that it is an unsupervised learning algorithm. The changes in the weights are only affected by local changes and the trained neurons exhibit an orientation selectivity. As the learning proceeds, the weight vector moves progressively closer to the direction of maximal variance in the data.

- **Competitive learning rule**. In this type of learning, all the neurons are forced to compete among themselves such that only one neuron will be activated and this neuron is the only one that will adjust all its weights. This phenomenon is known as winner-take-all. Competitive learning has its inspiration in the biological nervous system and often consists of clustering or categorizing the input data based on data correlations. Similar patterns are stored and represented by one single neuron. Competitive networks use inhibitory connections so that the winner output inhibits the other neurons to fire, and a simple competitive learning rule can be stated as

$$\Delta w_{ij} = \begin{cases} \eta \cdot (x_j - w_{ij}) & i = i^*, \\ 0 & i \neq i^*. \end{cases} \tag{3.30}$$

where, i^* is the winner neuron. Note that only the weights of the winner unit get updated. The effect behind of this learning rule is to store the pattern in the winner neuron through the weights that tend to be closer to the input pattern. As in other learning rules, this rule will never stop learning unless the learning rate is $\eta = 0$. This problem is related also with the stability of the system, and one way to achieve it is to force η to decrease gradually with the time of the learning. However, this solution causes another problem related with the plasticity of the network and its ability to adapt to new data. One of the most well-known examples of competitive learning is vector quantization for data compression (Jain et al., 1996).

The most used learning rules are the error-correction ones, specifically the ones related with gradient descent error learning, such as backpropagation because with enough time the algorithm converges to the solution if it exists. Moreover, for the majority of applications, the information about the desired output is available. Thus, it is known how the ANN has to behave and a supervised learning algorithm can be used. But to make learning successful, it is necessary to consider some aspects of the algorithms and the data used.

3.5.1.2 Concerns

There are some concerns related to the application of the learning algorithms and how to apply them successfully into the problem at hands. First of all, three practical issues have to be addressed, which are associated with the learning theory, specifically learning from samples: capacity, sample complexity and computational complexity (Jain et al., 1996). The first one is related with the ANN structure chosen to solve the problem. Thus, the capacity is related with the number of patterns that can be stored in the network, the function that the network will perform and the decision boundaries among the neurons. It is necessary to choose wisely the learning algorithm together

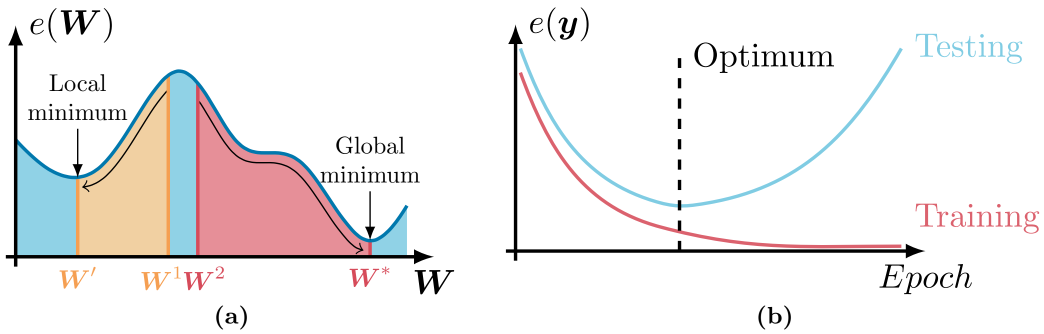


Figure 3.9: Learning algorithms concerns during the training of ANNs: (a) initializing synaptic weights and (b) stopping criteria based on the performance of the ANN.

with the structure and the application because it can happen that the problem does not have a solution for the combination. For example, underfitting problems typically occurred when the error of the network performance is too high, and they could be caused by using a too simple ANN architecture.

The next issue to take into account is the data used during the ANN training phase. Normally, the available data for the development of the ANN have to be divided in three datasets: training, test and validation. The training dataset is used during the weight update phase of the network. So, it is necessary that the samples of the training dataset are complex enough to guarantee that all the possibilities are presented to the network during this phase. A well-composed training dataset guarantees a valid generalization. The test dataset is used during the training of the weights to check the network response for untrained data and if the ANN needs more training. It has different examples from the ones of the training dataset but with the same characteristics. The validation dataset contains the rest of the data and is composed of data different from the other two datasets. It is used when the learning process has finished testing the abilities of the network with data that were never used during the learning process.

There is not a rule to follow in order to divide the dataset. It is a trial an error process. However, a large test dataset may help to obtain a better generalization capability, but the remaining datasets may not have enough data. Particularly, an appropriate division should be 40–50 % for the training dataset, 30–40 % for the test dataset and a 10–20 % for the validation dataset. With an appropriate division of the dataset, problems related with the underfitting of the network and also the overfitting problem could be avoided. This last problem is related with having a low training error but a high test error of the ANN performance, highlighting that there is not enough data for training it or that the structure is too complex.

It is also very important that the data have the proper form and is adequate to solve the problem during the training that is why they are normally preprocessed to accelerate convergence. For example, some techniques used are: noise removal, reducing input dimensionality, data transformation, data inspection and deletion of outliers. Other important issue related with data is that they have to be balanced to avoid the over-representation of one class. Moreover, it has to contain a large amount of data to train successfully the network. Sometimes the database of the network is expanded by adding noise to the available data in order to obtain new examples when new ones cannot be obtained.

Another concern related with ANNs is the weight initialization because it can affect to the algorithm convergence. For example, in Figure 3.9(a), it can be observed two different initializations of the weights $\boldsymbol{W}^1$ and $\boldsymbol{W}^2$. In gradient descent learning algorithms, such as the backpropagation, the first situation leads to a suboptimal solution because the algorithm is trapped in a local minimum. Whereas in the second situation, it achieves the optimal solution because it reaches a global minimum. Thus, it is very important to select the right initialization for the convergence of the algorithm since the appropriate solution can be found and the speed of convergence to the solution is accelerated.

In order to achieve a good network performance, it is necessary to take into account some elements of the training configuration. One of the factors to take into account is the learning rate (η), which is related with the speed of the learning algorithm. If η is high, it will accelerate the training because the changes in weights are high but it can cause the search to oscillate on the error surface and never converge. On the contrary, a small value will make the search steadily in the direction of the minimum error, but slowly. Normally, this value is constant and its value is $\eta \in (0, 1.0]$.

Another important parameter of the learning is the convergence criteria used to stop the training. Normally, three criteria are used: training error ($e \leq \varepsilon$), gradient of error ($\nabla e \leq \delta$), and cross-validation (a combination of the two previous). In all the cases, the network will not stop until the performance is reduced to a minimum. However, some problems are found such as in Figure 3.9(b) in which the test error is growing while the training error continues decreasing its value (overfitting). In this case, it is better to stop the learning because the optimum network performance was found when both error were minimum. This could also be included in the convergence stopping criteria.

Another parameter that affects a learning algorithm is the number of epochs used to train the network. This number has to be large enough to guarantee that the minimum error is reached during the training and the ANN has reached a steady state in which it cannot improve more its performance. All these parameters are related with the computational complexity of the algorithm, which refers to the time required for a learning algorithm to estimate a solution from training patterns. Therefore, a learning algorithm has to be well configured in order to obtain the right training for the network in time.

3.5.2 Tuning: Genetic Algorithm

In this Section, another way to train ANNs is introduced and it is based on Genetic Algorithms (GAs). GAs are probabilistic search or optimization algorithms based on genetic mechanisms of natural selection and inspired in Darwinian principles. From a theoretical point of view, these are global optimization algorithms, such as simulated annealing (Kirkpatrick, 1982) and tabu search (Glover and Laguna, 1999).

A GA consists of transforming iteratively a set of mathematical objects called *population*, each one with a cost value or *fitness value* associated, in a new population of descendants using biological genetic operators such as selection, crossover or mutation (Golberg, 2006). The *fitness function* corresponds to the optimization function and it is the one that assigns the cost to each one of the individuals. The higher the fitness value of an individual is, the better solution to the problem is.

The main interest on this type of algorithms, like other metaheuristics, is based on the small number of constraints on the function to be optimized. Particularly, this has not to be differentiable, unlike gradient based methods of learning algorithms. In addition, GAs are global search methods, so that the problem of some learning algorithms of getting stuck in local minima disappears. A GA can search for the optimum solution in the entire space of solutions to find it. Thus, GAs make efficient searches in complex spaces and they are also computationally simple but at the same time powerful because they are not limited by restrictive conditions of the search space (continuity, differentiability, etc.) (Golberg, 2006).

The GAs have been used in applications related with optimization or Machine Learning (ML). In this Thesis, GAs are used to adjust the free parameters of an ANN structure based on a fitness function. In this way, the GA is used to optimize the structure of the ANN. In the GA context, optimization is understood as an improvement process that will always get to improve the previous situation. GAs are characterized by (Golberg, 2006):

- Working with a set of encoded parameters in a string of finite length over a finite alphabet (chromosome).

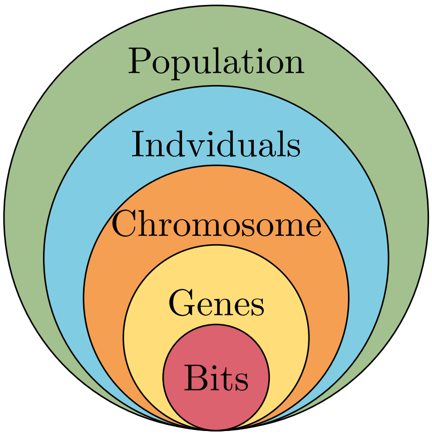


Figure 3.10: Hierarchy of the elements that composed the population of a GA.

- Using a population of individuals, in this way it offers a view of the whole rather than a single point. Therefore, search is performed in parallel.
- Using only a fitness function or cost information associated with each individual, without using any other knowledge and focusing on finding better individuals.
- Using probabilistic rules for transitions between generations. The transition rules are stochastic operators.

The bioinspiration of this algorithm is obvious, it can be appreciated in the analogy of the nomenclature used to describe the different strings that compose the individuals which are called chromosome. The set of chromosomes is referred as the genotype and the phenotype is the genotype together with its environment. A chromosome is composed by a string of values, each one called gene and represented by bits, the value is called *alleles* and the position that it occupies in the chromosome is the *locus*. The GA is responsible for transforming this population of individuals in another population of descendants by using some genetic operators. Thus, for a better understanding, the operation principles of the GA are described in Section 3.5.2.1 and some genetic operators are described in Section 3.5.2.2.

3.5.2.1 Operation principle

In this Section, it is explained how a general GA works. As mentioned before, a GA works on an initially random population of individuals and each of them represents a solution of the problem in which is applied. Each solution is represented by a genotype (a set of chromosomes) and interpreted on the form of a phenotype, which consists of the visible behavior of the individual. Finally, it is necessary to define an adaptive or cost function, known as fitness function, that evaluates the performance of each individual in the population. The more efficient are chosen to solve the problem. The individuals with higher fitness values will have a bigger chance to let their offspring genotype.

Each solution parameter corresponding to the chromosome of an individual is assimilated in a gene. A chromosome is a string of genes which can consist of similar parameters from the same chromosome. In addition, each gene is reachable by its position or locus on the chromosome. Each individual is represented by a chromosome or set of chromosomes and a population is a group of individuals. Figure 3.10 represents the different elements of a population and the hierarchy between them. There exist different forms to encode the genes of the chromosome. For example, genes can be encoded by using a binary codification, in which each gene is formed with a value of 0 or 1. On the other hand, the genes could have a real value to form the chromosome so a real codification is used. A special codification is the use of the

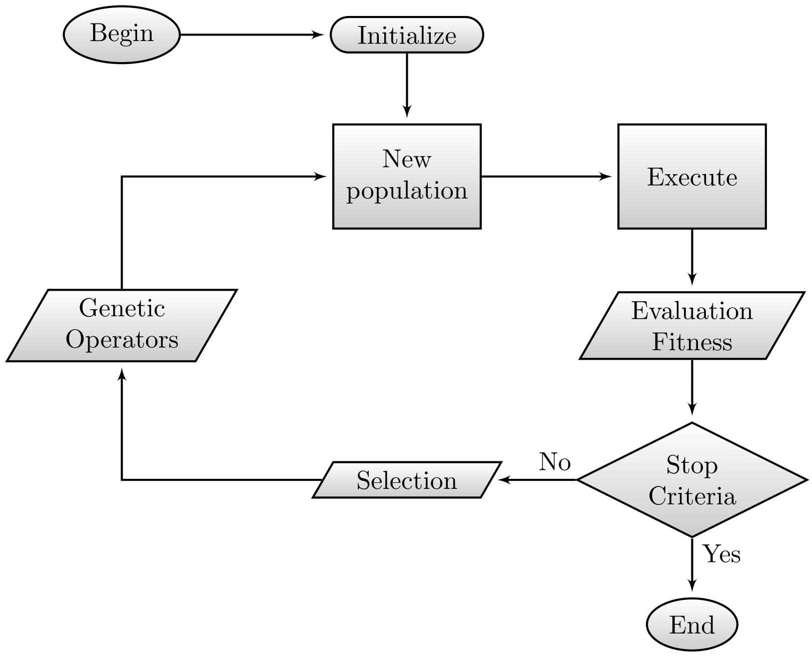


Figure 3.11: Flow diagram of the behavior of a general GA.

Gray codification, in this case the Hamming distance is used to measure the similitude between two elements of the population, so that between two neighboring elements in the search space, one bit differs.

Figure 3.11 shows an overview of the implementation of a GA in general. Executing a GA consists of the following parts:

- The algorithm is initialized setting up all the parameters of the environment such as the number of individuals per generation, genetic operators parameters, the fitness function, the number of generations to reach a solution, etc.
- Then, a new population is created, firstly using random values for the chromosomes of the individuals.
- Once the population is created, each individual is executed for the problem conditions in which the algorithm is applied.
- Then, the performance of each individual is evaluated, a cost is assigned through the fitness function and the individuals will be ordered by cost from the best individual to the worst.
- After that, it will be evaluated if the stopping criteria is reached. If it is reached, then the algorithm will finish and the best chromosome is found to solve the problem. Otherwise, the algorithm will continue preparing a new generation until it reaches the stopping criteria.
- Then, the algorithm selects the best individuals who will serve to form the next generation and their chromosomes prevail for further improvement.
- Some genetic operators are applied to the best individuals to generate its offspring that will be part of the next generation and the rest of the individuals are discarded.
- Finally, a new generation is created and the cycle is repeated until the stopping criterion is matched.

There are some considerations that have to be taken into account when using GAs. The core piece of the GA is the fitness function. When using a GA, it is necessary that this function is well defined with respect to the problem, otherwise the GA will not be able to solve it or will have difficulty converging at all. The design of a fitness function is a complex process, so that it is normally used in applications where the problem is well identified. Moreover, the fitness function evaluation is related with the time that the algorithm requires solving the problem. Thus, for complex problems, the fitness evaluation is large and it is limiting the algorithm. In addition, GAs with a fitness function with abrupt changes between two states, such as in decision problems, are not able to converge to the solution because of the oscillation in the solutions.

The stopping criterion is another issue of GA. Normally, the GA will be working generation after generation until it finds the best possible solution of the problem and obtains the highest value of the fitness. However, a stopping criterion is usually used based on the stuck value of the fitness during a number of generations in combination with a maximum number of generations. Finally, selection and other operators of the GA are presented in Section 3.5.2.2.

3.5.2.2 Genetic operators

Three main genetic operators are: selection, crossover and mutation. They are applied between generations to form the next one. First the individuals are selected according to their fitness value, then the crossover is applied between chromosomes and finally, the mutation is in charge of altering some genes.

The *selection* operator is responsible for defining which individuals of the current population will be used in the new population of the next generation, which will serve as parents during the crossover operation. This operator can be considered the most important because it allows individuals in a population to survive, reproduce or die. In general, the probability of survival of an individual is directly related to their relative effectiveness within the population, i.e. its fitness value. Four types of selection methods are introduced (Golberg, 2006).

- *Roulette-wheel selection.* This method is the most used. With this selection method, each individual has a chance of being selected proportional to its performance. Thus, the fittest individuals are the ones that have higher probabilities to be selected to breed the next generation. Imagine a roulette or wheel, each individual is associated with a sector whose size is proportional to its fitness value. This selection has some drawbacks because of the variance that presents when selecting the individuals that compose the wheel. For example, in n selections, the same individual can be selected to be a parent of the new generation and no alteration of the fitness value for the new individual will be reached, leading to a loss of diversity. Moreover, it could also fall in individuals with low fitness in spite of the low probability. Therefore, this selection is normally combined with other types, to avoid these problems.

- *Elitism.* This method consists of selecting the best n individuals, which have the higher values of fitness, to be part of the new population. It is usually combined with the previous selection method to avoid some of the aforementioned problems.

- *Tournament selection.* In this method, two individuals are chosen among the base population, making them fight. The individual with the highest fitness has a probability p to win the fight defined between 0.5 and 1. This process is repeated until all the individuals of the new generation are selected. Again the variance of this method is high, and the fact that the probability p increases or decreases its value, allows increasing or decreasing the selection pressure.

- *Stochastic universal sampling.* It is a technique designed to map the original fitness value of an individual with its expected fitness value, so that the GA is less susceptible to premature convergence. This selection consists of distributing the population in a segment based on the fitness values of its individuals, and then selecting them based on a set of equidistant points.

The next genetic operator used after the selection is the *crossover*. This operator consists of creating a new offspring from the individuals of the selections that are used to breed the new individuals of the next generation. Hence, the new individuals after the crossover inherit partially the characteristics of their parents. This operator is applied to a couple of individuals and it is made in two steps. The first one consists of selecting the individuals and associate them randomly in couples (discarding the elite). And then the crossover occurs in one, two or more points defined on the chromosomes randomly, so each one is separated in different segments. After this, each segment is swapped with the corresponding segment of the other parent with a chance of crossing defined. Thus, it can be noted that the number of crossing points and the probability of crossing, provides more or less diversity in the GA. An example of the crossover operator is as follows, imagine that there are two parents x_1 and x_2 with a chromosome length of five genes.

$$x_1 = 10|011$$

$$x_2 = 01|110$$

where the crossing point is $k = 2$ and the new individuals for the next generation are:

$$x'_1 = 10110$$

$$x'_2 = 01011$$

This is the simplest crossover operator but other crossover operators more advanced are (Golberg, 2006):

- *Partially matched crossover (PMX)*. This type of crossover consists of selecting two random points inside the chromosome to establish the crossing area of both chromosomes. After that, with the parts of the chromosome that are not involved in the crossover a permutation of the genes is established. For example, suppose that,
$$x_1 = 9\ 8\ 4 \mid 5\ 6\ 7 \mid 1\ 3\ 2\ 10$$
$$x_2 = 8\ 7\ 1 \mid 2\ 3\ 10 \mid 9\ 5\ 4\ 6.$$
Then, the central block is swapped and in the other two blocks per chromosome, the genes are permuted, obtaining
$$x'_1 = 9\ 8\ 4 \mid 2\ 3\ 10 \mid 1\ 6\ 5\ 7$$
$$x'_2 = 8\ 10\ 1 \mid 5\ 6\ 7 \mid 9\ 2\ 4\ 3.$$
- *Order crossover (OX)*. This operator establishes and area for swapping the genes and then it changes them. Unlike the previous operator, in the areas that are not exchanged instead of being permuted, the chromosome rearranges itself. For example suppose that,
$$x_1 = 9\ 8\ 4 \mid 5\ 6\ 7 \mid 1\ 3\ 2\ 10$$
$$x_2 = 8\ 7\ 1 \mid 2\ 3\ 10 \mid 9\ 5\ 4\ 6$$
when x_2 makes the exchange of the values, these values are empty and there is not any value to fill them.
$$x_2 = 8\ -\ 1 \mid 2\ 3\ 10 \mid 9\ -\ 4\ -$$
and now the chromosome is reordered, putting together the empty spots in the swapping area,
$$x_2 = 2\ 3\ 10 \mid -\ -\ - \mid 9\ 4\ 8\ 1$$
finally, it is obtained that,
$$x'_1 = 5\ 6\ 7 \mid 2\ 3\ 10 \mid 1\ 9\ 8\ 4$$
$$x'_2 = 2\ 3\ 10 \mid 5\ 6\ 7 \mid 9\ 4\ 8\ 1.$$

- *Cycle crossover (CX)*. This type of crossover is different to the two previous ones since a swap area is not established. In this case, a point of the chromosome is chosen from which there will be no changes at all. Then, the value of the first gene of the second chromosome is sought. If the value of each gene for the two chromosomes does not match, it has ended and it would change all the genes of both chromosomes. Otherwise, it would set the first chromosome gene whose value is equal to the first gene of the second and would repeat the operation. An example of this operator is as follows,

$$x_1 = 9\ 8\ 1\ 7\ 4\ 5$$

$$x_2 = 1\ 2\ 4\ 5\ 9\ 7$$

first, it is taken the first gene of x_1

$$x_1' = 9\ -\ -\ -\ -\ -$$

the value of the first gene of x_2 is 1 which is also in x_1', so that it will be fixed in the chromosome,

$$x_1' = 9\ -\ 1\ -\ -\ -$$

the value in x_2 corresponding to the position that it has been fixed in x_1' is 4, so it will also be fixed,

$$x_1' = 9\ -\ 1\ -\ 4\ -$$

finally, the value of the gene in x_2 is 9 that is already fixed so the cycle is finished and now the exchange of genes can start,

$$x_1' = 9\ 2\ 1\ 5\ 4\ 7$$

$$x_2' = 1\ 8\ 4\ 7\ 9\ 5$$

The crossover operator favors the exploration of the search space. Considering two genes A and B that can be improved by mutation, it is unlikely that the A' and B' genes altered by the mutation appear in the same individual. But if one parent has the gene A' and the other B', the crossover operator will combine the two genes, and thus create a new individual by taking this combination. With this new combination it is possible that the new individual is more adapted than their parents. The crossover operator allows the mixture of genetic material and accumulation of favorable mutations. But it is possible that the joint action of selection and crossing forbids the convergence to the optimal solution of the problem. Thus, considering a population of individuals having a chromosome with a single gene with binary values, if none of the individuals in the initial population has the value 1 for this gene, selection and crossover will not allow the appearance of the other values.

It is in this scenario where the *mutation* operator arises. This operator consists of changing the value of a gene randomly with a very low probability to do it. In the case of binary codification of the chromosome, the mutation only inverts the value of the gene in a random location of the chromosome. Thus, mutation modifies randomly the characteristics of the solution and it allows introducing diversity in the solution population. Mutation can be interpreted as a noise that interferes with the solution by adding new points in the search space. This operator has three main advantages. The first one is that this operator avoids the phenomenon of genetic drift, when some genes favored by fortune are used to the detriment of others, and are thus present in the same place in all chromosomes. It also avoids the risk of the premature convergence of the algorithm in which all the individuals are exactly the same and imitates local optima solutions because of the diversity that the operator introduces. And the last one is that it guarantees the ergodicity property in which each point of the solution space can be reached. Random mutations in all positions of the gene mathematically guarantee that all the solutions are collected in infinite time, so that it is guaranteed to find the global optimum.

Therefore, a simple GA typically consists of a selection, a crossover and a mutation operator to be able to explore the entire solution space. Other complex operators can be added to GA. For example, *diploidy* consists of having one or more couples of chromosomes, each one of them has information of the same function and will be used to be crossed. The *dominance* is another operator that is used to eliminate redundancies between two equal chromosomes so that preference is given to a particular value against other values that the gene can take. *Inversion* consists of choosing two points inside the chromosome which is cut at such points and the positions of the ends of the cut section are exchanged. In *segregation*, during the crossover, only one of the parents is chosen to be part of the next generation. And *translocation* which is a crossover inside the chromosome, in which the genes location is changed.

To sum up, different GA operators have been described in order to explain how the creation of the next generation is carried out. There are some operators more complex than others, but all of them are applied to obtain the best individual that solves the problem in the fitness function way. In general, not all the operators are used for the same GA. As mentioned above, the most common GAs are composed by the three main operators, selection, crossover and mutation. In this Thesis, a GA is going to be used to optimize the structure of the ANN selected and to that extent, the genetic operators used are the basic ones in order to simplify the problem of training the ANN.

3.6 Conclusion

In this Chapter, a deep literature review of the Artificial Neural Networks (ANNs) has been done in order to understand better the advantages and disadvantages of using this type of AI algorithms in the management of a grid. In first place, a historical review was done to understand the evolution of the ANNs and the different milestones involved in their development. Nowadays, the ANNs are commonly used due to their appealing properties of distributivity, robustness, generalization, redundancy, adaptivity, etc. One of the main reasons of their intensive use is the capacity of extracting features for large amount of data, which is a worrying problem in the actual society as more data is gathered from the individuals. ANNs fit perfectly for Big Data Analytics (BDA) and Machine Learning (ML).

The different features of the ANNs were presented together with some important architectures and types. Of particular interest are the Recurrent Neural Networks (RNNs) since they present some dynamics characteristics that help in the construction of an algorithm that manages the power flows of the grid. Therefore, a part of the Chapter was dedicated to understand these type of networks and the stability properties that inherit their structure. The reasons for using them are the presence of a short memory in the feedback loops of the connection structure and the possibility to model the dynamic behavior of any system.

After describing the most famous architectures and the ones used in this Thesis, it was necessary to introduce the different fields of applications. ANNs are used mainly in classification, data processing, function approximation, optimization, association and control. However, ANNs can be applied almost to any problem that it is well defined. Moreover, there are different applications in which ANNs are applied, such as power systems. Some of these applications are: load forecasting, system security, transient stability, fault diagnosis and economic dispatch.

Finally, it was necessary to explain how ANNs are trained in order to fulfill the task in which they are applied. Two methods have been explained, learning and tuning. Traditionally, learning algorithms have been used to train ANNs by changing the free parameters of the network. Depending on the application and the available data, there exist different approximations to use during the training: supervised, unsupervised and reinforcement learning. In addition, other classifications were introduced to understand different aspects on how the ANNs learn from the environment. However, these algorithms present some concerns such as the initialization of the structure or finding suboptimal solutions to the problem. On the other hand, the Genetic

Algorithms (GAs) are used to optimize the ANN structure based on the minimization of a fitness function defined by the user. In general, three operators are used to obtain the new individuals of the next generation: selection, crossover and mutation.

In this Part of the Thesis, the general concepts of the electrical grid were introduced such as the forms of generation, the transport and distribution of electricity, the different consumptions inside the grid, etc. In addition, some problems with the actual status of the grid were introduced in order to tackle them with the upcoming arrival of the Smart Grid (SG). However, there exist barriers to the SG deployment, one of them is related with the absence of algorithms to manage the grid. The needs of the grid will be attended by SG, but it is necessary to develop algorithms that manage not only the power flows but also the information coming from the different participants of the grid. Thus, a possible algorithm, the ANNs, was detailed for its use in this Thesis with the objective of managing the grid power flow from the demand side making possible a SG scenario. Part II develops the solution found to control the power flows from a DSM approximation using the concepts introduced in this first Part of the Thesis.

PART II

NeuralGrid: An approximation towards Smart Grid

4 Individual Controller

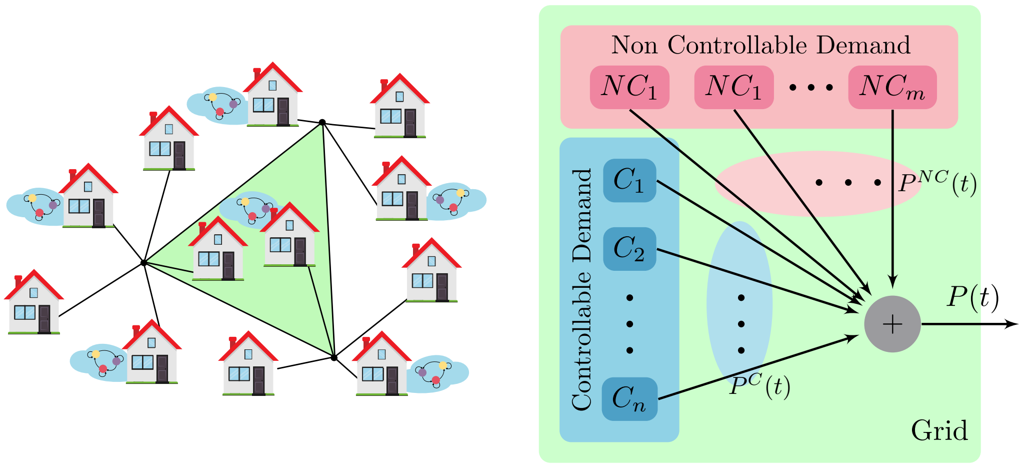
"Everything should be made as simple as possible, but not simpler." — Albert Einstein

Working with Artificial Neural Networks (ANNs) in power systems has been long used. However, their application field is centered in tasks related with their pattern recognition and forecasting abilities, in order to handle problems that arise in the grid. In this Part of the Thesis, the use of ANNs is proposed to manage the local demand of different grid elements to smooth their aggregated consumption. However, it is done from a particular point of view focusing on Low Voltage (LV) users instead of using the capabilities of cutting large grid consumptions. In this way, the security of supply is guaranteed for all users and the operation enhancement of the entire grid.

As mentioned in Section 2.3.4, there exist different Demand Side Management (DSM) techniques to address this issue. They are implemented in the form of different incentives such as energy savings due to the temporal displacement of loads or grid status regulation through the direct load control. The DSM algorithm proposed in the Thesis is based on the load automation of the local electric behavior of LV users. The followed DSM strategy is applied locally, but its effects impact globally in the system. Thus, the different users within the grid are self-organized to contribute to smooth their aggregated consumption. The implementation of this distributed DSM strategy is the leitmotiv of this Part of the Thesis. A neural approach is used to implement it which consists of controlling the local electric behaviour by taking into account the aggregated consumption of the electrical grid, local energy resources and user requirements.

The use of ANNs allows applying their abilities of signal processing and forecasting to vary the behavior of the grid aggregated consumption. Moreover, their adaptive and distributed properties make it perfect to implement a possible solution to this electric paradigm. Specifically, Recurrent Neural Networks (RNNs) are used to tackle this problem, due to its dynamic properties and its short-term memory ability inherited in its structure. Both are desirable properties to design a DSM controller that has to adapt to changes in the environment. However, working with RNNs can be difficult and computationally slow, especially for large structures. Therefore, a small RNN structure is sought in order to be implemented in each user, by building a modular system of neural blocks that can be executed no matter the technology. The neural controller designed is going to extract some features from its inputs by using temporal series analysis and adaptive filtering theory.

The interest of developing a DSM algorithm is motivated to find solutions to some problems of the current electrical grids and as a possible solution to improve the integration of the next generation of grids or Smart Grid (SG). Thus, the idea is to use the data coming from the current monitoring platform, which is increasing their capabilities thanks to the deployment of smart meters, and the local information available from the user. There is no possibility to share information between users because the grid does not allow it and guarantees the anonymity of its users. Therefore, a solution to the problem is elaborated with the minimum information possible. It must be guaranteed the integrity of the data coming form the users and the grid operation enhancement. Herein, lies one of the problems that will be tackled



(a) (b)

Figure 4.1: Formulation of the environment in which ANNs act: (a) graphical representation of the different elements that compose the environment and (b) its translation to a block diagram. The houses with a blue cloud represent the controllable users and the rest are non-controllable.

during the development of the neural controller on this Thesis, since ANNs work better the more information they possess from the environment.

However, the problem cannot be addressed directly since the complexity of the grid environment. Firstly, the environment and the different elements inside it are introduced in Section 4.1. In order to develop a first approximation of the solution, the environment was simplified (see Section 4.2). The different parameters involved in the training of the neural controller and how is carried are explained in Section 4.3. Then, the results achieved during the training are presented in Section 4.4. In Section 4.5, a post-evaluation is carried out to test the neural controller achieved at the end of the training. Finally, the conclusions of the Chapter are gathered in Section 4.6.

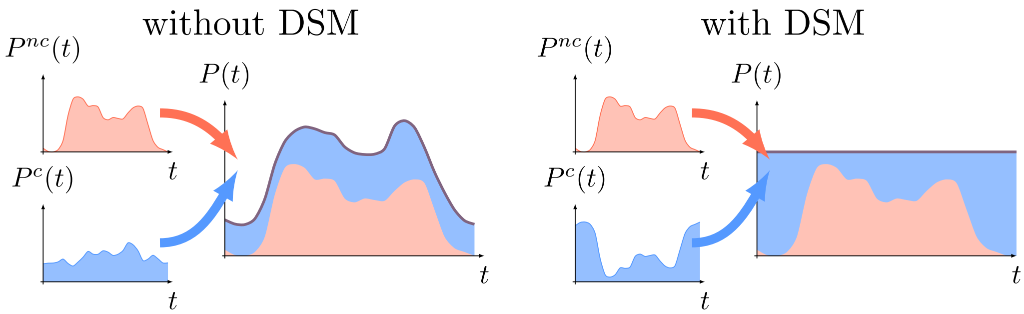
4.1 Environment

In first place, to solve a problem with ANNs, it is necessary to know and understand the environment in which they will be deployed. Thus, the environment of the problem consists of LV users, specifically users from the residential sector, whose consumption is typical for a high electrified user (see Section 4.1.1). The electricity required by those users will be supplied by the grid. The grid is composed by two types of users: i) non-controllable users, whose consumption cannot be controlled because it is not deferrable or they do not have any DSM system, and ii) controllable users, who can control their demand in real time through the use of a controller. Figure 4.1(a) shows a representation of the grid with the two user types in which the grid is divided.

The aim of the neural DSM controller is to flatten the aggregated consumption of a grid composed by this two type of users. Mathematically, if $P(t)$ is the formulation of the aggregated consumption, then the objective of the neural DSM controller consists of $P(t) \rightarrow C$, where C is a constant. Moreover, $P(t)$ can be divided in the sum of the two users of the environment, such that:

$$P(t) = P^{nc}(t) + P^{c}(t) \tag{4.1}$$

where, $P^{nc}(t)$ is the non-controllable consumption and $P^{c}(t)$ is the consumption available for the algorithm to be controlled.



(a) **(b)**

Figure 4.2: Graphical representation of $P(t)$ signal as a sum of $P^{nc}(t)$ and $P^c(t)$ signals, where: (a) represents the current electric grid where $P^c(t)$ is not adapted to $P^{nc}(t)$ and (b) represents the proposed algorithm goal where $P^c(t)$ is adapted to $P^{nc}(t)$ in order to smooth $P(t)$.

Figure 4.1(b) shows a block diagram of the whole system in which the grid has been divided. These two blocks correspond to the non-controllable and the controllable demand, which at the same time are subdivided in m and n parts respectively. Thus, each part of the demand, $P^{nc}(t)$ and $P^c(t)$, is represented by the sum of the power consumed by each type of user (see Equations 4.2a and 4.2b).

$$P^{nc}(t) = \sum_{i=1}^{m} p_i^{nc}(t) \qquad (4.2a) \qquad\qquad P^c(t) = \sum_{i=1}^{n} p_i^c(t) \qquad (4.2b)$$

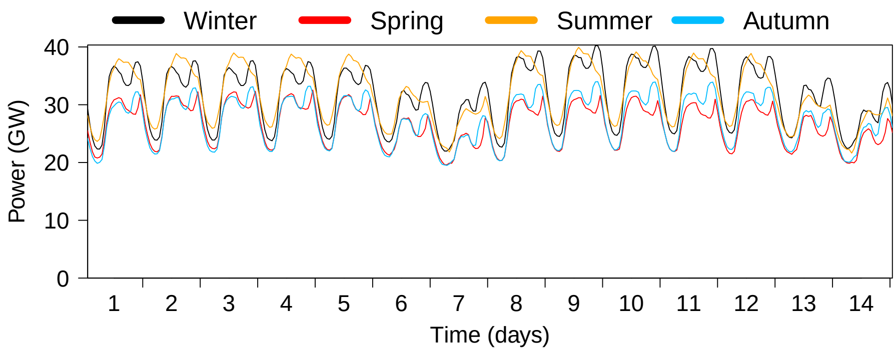
where, $p_i^{nc}(t)$ is the demand of the ith non-controllable user and $p_i^c(t)$ is the consumption of the ith controllable user.

The electrical behavior of the users and their aggregated consumption is described in Figure 4.2. In the general case, neither of the users can control their demand so that each one of them can consume any power at any time as in Figure 4.2(a). In this case, there is no algorithm to control the demand of $P^c(t)$ and the aggregated consumption possesses a high variability. Thus, it is necessary for the proposed algorithm to reshape the amount of $P^c(t)$ in order to adapt to $P^{nc}(t)$, achieving a flattened $P(t)$ consumption as in Figure 4.2(b). The main challenge, in the design of this type of algorithms consists of how to adapt $P^c(t)$ to $P^{nc}(t)$ to produce a flattened response. However, the current information coming from the grid and available by users is non-existent, making difficult the elaboration of a control strategy in order to achieve any goal. Hence, it is necessary to use some techniques already developed in other fields such as Artificial Intelligence (AI) or signal processing to extract some useful information from the small amount of information provided by the grid.

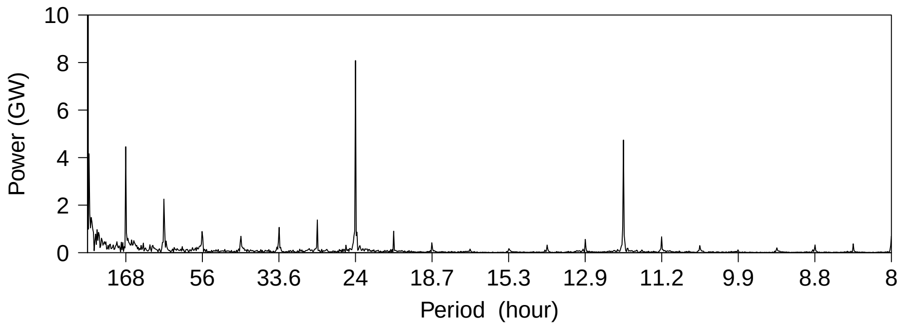
Information coming from the grid differs from one country to another. Normally, the transmission system operators possess all the information about the demand. For example, in Europe the information about the aggregated consumption of different countries can be found in European Network of Transmission System Operators (ENTSO-E)[1], which represent 41 electricity transmission system operators from 34 countries across Europe. In a grid in which users do not control their consumption, the aggregated consumption behaves periodically as shown in Figure 4.3. The data to elaborate Figure 4.3 was obtained from the aggregated consumption of Spain during 2015[2]. The aggregated consumption of other countries may differ in the form or in the power consumed, but the periodicity of valleys and peaks is very similar among them because consumption habits among users are very similar. Furthermore, it can

[1] https://www.entsoe.eu

[2] *Source: Red Eléctrica de España (REE)*, the transmission system operator of Spain.



(a)



(b)

Figure 4.3: Aggregated consumption of the Spanish grid during 2015: (a) temporal representation of two weeks for different seasons of the year and (b) annual spectrum.

be observed that the grid aggregated consumption of the same region is similar over the years, but with very small variations.

Focusing on the grid example of Figure 4.3, it shows a periodicity both in time (see Figure 4.3(a)) and in frequency (see Figure 4.3(b)). Figure 4.3(a) shows the aggregated consumption of two weeks in different seasons of the year. The behavior of weekdays is that they are very similar to each other and to other weekdays from other weeks during the same weather season. The same behavior can be observed for weekends. There are also similarities between winter and summer days and the same for spring and autumn days. This is also corroborated with the annual grid consumption spectrum of Figure 4.3(b), in which the most significant components are shown. It can also be observed that the strongest component (apart from the continuous one of period 0, corresponding to the average consumption) is the one located at a period of 24 hour or one day. The next stronger components are located at 12 hour period (half day) and 168 hour period (one week), respectively. Thus, it is necessary to reduce these components to smooth the aggregated consumption.

One possibility to eliminate these undesirable frequencies consists of displacing the consumption from those components to the 0 component. This approach was developed in Castillo-Cagigal (2014), in which by applying signal processing tools, such as the Fourier transform and Swarm Intelligence (SI) techniques, was able to synchronize a collective of controllable users with non-controllable ones through DSM in facilities with local Photovoltaics (PV) generation. The algorithm used in Castillo-Cagigal (2014) achieved the objective of enhancing the grid through the smoothing of the demand curve from the frequency domain.

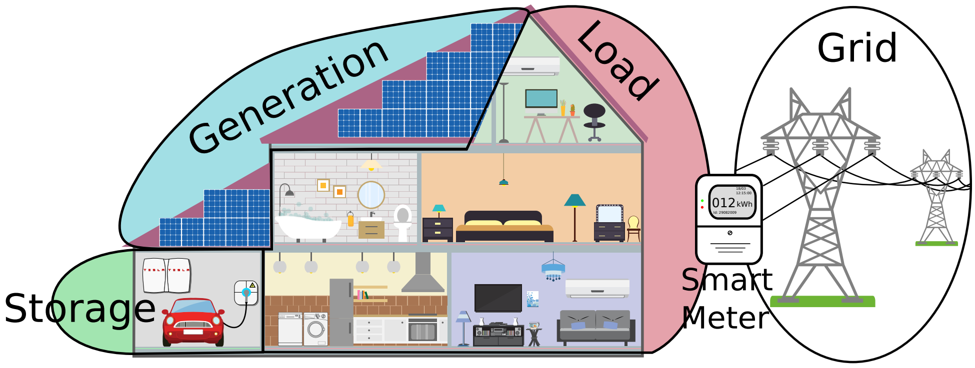


Figure 4.4: Controllable facility connected to the grid.

But, is it possible to design an algorithm following a time domain strategy to adapt $P^c(t)$ to $P^{nc}(t)$ in real time? This question is going to be solved along this Part of the Thesis, using the properties of ANNs. Nevertheless, before describing the algorithm strategy followed, it is described what is inside of each user of the environment to know its consumption profile. Section 4.1.1 describes the facilities that are inside the artificial grid developed for this Thesis.

4.1.1 Facility

An electrical grid is composed by a large variety of consumptions to improve their performance through maximizing the utilization of the installed capacity. Therefore, different consumption profiles are found inside them, such as industrial factories, commercial users or residential ones. The grid is managed by operators in charge of the different parts that comprise it. However, the load consumption is part from the local electric power system of the user. In order to simplify how these power systems are addressed, they will be referred as *facilities* along this Thesis. A facility is owned by a particular consumer whose management depends on him instead of the electric utility. A facility is characterized by different attributes, for example its size varies depending on the needs of the user, from a single family house to large factories or even a microgrid composed by a neighboring community.

In this Thesis, the facilities consider the possibility to actuate in their consumption and they are divided in two groups: controllable and non-controllable users (see Figure 4.1). The second one represents all those consumptions within the grid for which there is no information about their consumption except the one coming from the aggregated consumption of all of them. Thus, there is no possibility to know any individual information of their load profile. On the other hand, the controllable facilities are perfectly known and their individual consumption is completely specified knowing the instant power they are consuming. This information would be available in the near future thanks to the smart meters that are being deployed. The meter is used as the common interface between users and the grid, being the communication link between both of them and is located in the entrance of the facility at the end of the distribution network. Figure 4.4 shows the representation of the controllable facilities used in this Thesis.

Traditionally, facilities were considered as mere consumers. However, this concept has been modified in last decades with the inclusion of new technologies which are crucial in the deployment of SG. As shown in Figure 4.4 the facility is connected to the grid through an electric meter which serves as the exchange point between the grid and the user to measure the electricity consumed by the loads. Inside the facility, there are three main parts that compose it and they are: i) generation, ii) storage and iii) consumption. These concepts have been already introduced in Chapter 2. The technology used to implement the local generator of the facilities in this Thesis is the PV. This technology has been chosen due to several reasons, such as its renewable

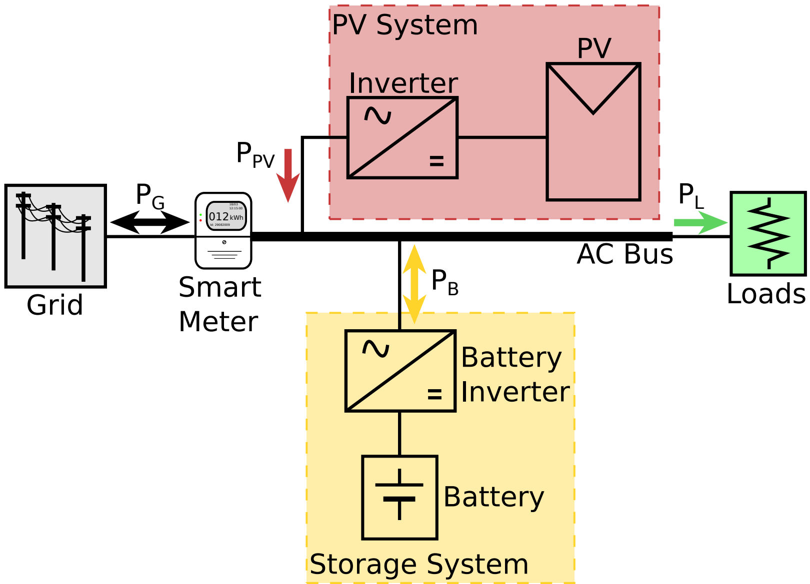


Figure 4.5: Schematic representation of the controllable facility, where the arrows represent the possible direction of power flows.

nature, its modular size of the generator, its ease integration in different housing types, etc. The size of the PV generator depends on the number of modules that comprises it.

However, a PV generator can only produce electricity during day hours, producing a bell shape electricity curve related with the radiation of the Sun. Therefore, the facility also incorporates an energy storage system for a better use of the energy generated locally. The incorporation of a storage system gives the facility a way to store the surplus of PV generated for a posteriori use. In spite of being expensive, a small storage system designed for only half a day of autonomy will increase drastically the local energy used to supply the loads (Castillo-Cagigal et al., 2011a). The storage system is used to complete the local energy source replacing the one coming from the grid. Thus, the grid is used as a mere back up source when the local energy is not available or not sufficient.

Finally, the facility consumption is the one of a highly electrified home which is composed by home electronics (TV, personal computers, media player, Hi-Fi equipment, etc.), lighting, appliances (washing machine, dryer, fridge, oven, dishwasher, etc.), air conditioning, etc. Among these loads, there are those which can be controlled and those which cannot. Hence, the consumption inside a controllable facility can be divided also into two groups depending on the controllable capacity of the load. And for each controllable facility the algorithm will dispose an amount of energy that can be controlled. The rest of the energy will depend on the user preferences.

But, how are the different elements of the facility related? Figure 4.5 shows a topology representation of all the elements that integrate the facilities used in this Thesis. The different parts of the facility are interconnected through a topology known as Alternating Current (AC) bus in which all the systems exchange energy in AC. So, the elements that use Direct Current (DC) such as the PV system and the battery needed an inverter to connect to the bus. This form of interconnection presents a lower performance of the generation system because it will be required to transform the energy generated from the PV system, i.e. from DC to AC and then AC to DC in order to charge the battery. In spite of the lower performance, the presence of inverters gives some other benefits, such as regulation of power factor, grid stabilization, reduction of the harmonic voltage distortion, etc. Furthermore, the losses that incorporate the AC bus are considered negligible. On the other hand, this

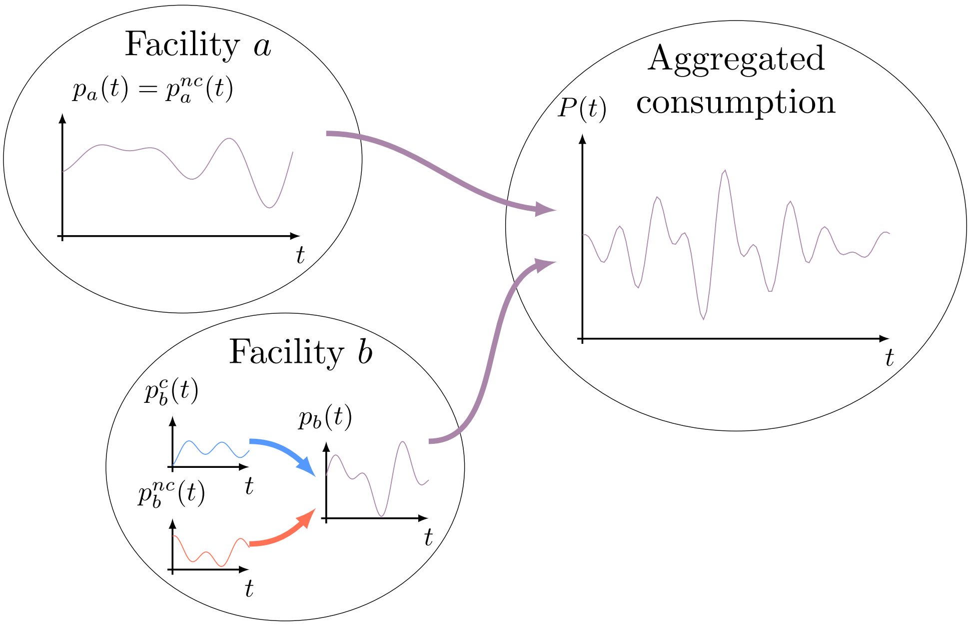


Figure 4.6: Grid representation divided in two types of facilities. There are two facilities a and b which are added to compose the aggregated consumption. Facility a is a non-controllable facility in which the demand cannot be controlled. Facility b is divided in two parts non-controllable consumption in red and controllable consumption in blue.

topology allows that the loads always have a source to be used in order to supply them. Thus, many possible exchanges are present among them. The electrical behavior of the facility responds to Equation 4.3:

$$P_{PV}(t) + P_B(t) + P_G(t) = P_L(t) \tag{4.3}$$

where, $P_L(t)$ is the instantaneous power consumed by loads, the $P_{PV}(t)$ is the instantaneous PV power generated, $P_B(t)$ is the instantaneous power exchanged with the local storage system and $P_G(t)$ is the instantaneous power exchanged with the grid. The sign of each variable depends on the power flow, for example $P_L(t)$ and $P_{PV}(t)$ are always positive. While $P_B(t)$ is positive when the local storage system supplies power to the loads and negative when it is storing energy. And $P_G(t)$ is positive when the grid supplies power to the loads and negative when the facility exports power to the grid. In this Thesis, it is considered that the only source to charge the battery is the surplus of generated energy and the only exported energy to the grid is the surplus of generated electricity when the battery is fully charged and the local demand is lower than the generated electricity.

In conclusion, the environment designed in this Thesis is composed by different facilities as it is shown in Figure 4.6. These facilities are divided in two types controllable and non-controllable. About the non-controllable ones, there is not enough information about them, the only known information is its consumption through the aggregated consumption of all the individuals. On the other hand, the controllable facilities are known and part of their consumption is controllable, making possible to alter their load profile by modifying the way some loads consume. However, there is no communication between the facilities. Thus, all of them receive the same information from the grid, the aggregated consumption. And with this information, the controllable facilities have to modify their consumption to adapt to the non-controllable ones. Section 4.2 describes the keys to begin the development of the algorithm that will manage the power flows of the grid to enhance its performance.

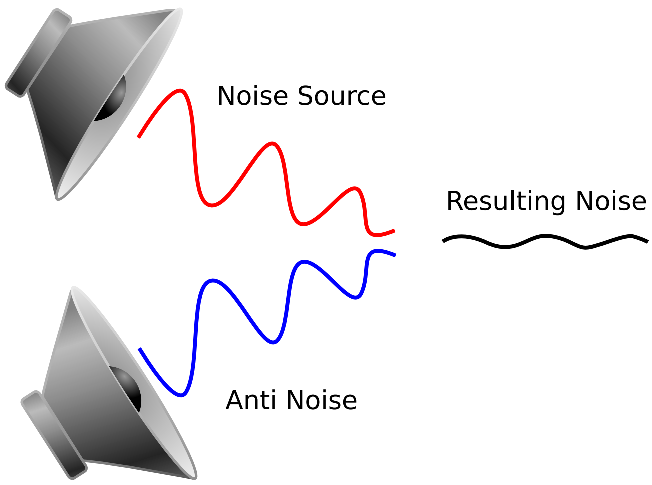


Figure 4.7: ANC principle described graphically.

4.2 Derivative algorithm

In order to enhance the grid, the proposed algorithm in this Thesis has to be able to adapt the power of the controllable facilities to the non-controllable ones without any information apart from the one of the aggregated consumption and the power they are consuming. Thus, there is no communication between the facilities of the grid. The objective of the algorithm consists of being able to meet the condition of Equation 4.4.

$$P(t) = \sum_{i=1}^{m} p_i^{nc}(t) + \sum_{i=1}^{n} p_i^{c}(t) = C \tag{4.4}$$

Then from Equation 4.4, it can be deduced that the result of applying the algorithm should be that,

$$\frac{dP(t)}{dt} = 0 \tag{4.5}$$

Equation 4.5 also implies that $P(t)$ is constant, but the different parts, that integrate it, are not necessarily constant. Therefore, the algorithm would modify the response of each of the $p_i^c(t)$ to adapt in real time to the $p_i^{nc}(t)$. In addition, it is known that the $P(t)$, with no active control of the facilities, presents strong periodicity components corresponding to the 12 h, 24 h and 1 week. Thus, a first possible approach would be to remove the power from those nonzero periods since the 0 period is equivalent to the average of the signal which has a constant form. On the other hand, there is another possibility consisting of neutralising $P^{nc}(t)$ through the construction of an interference signal ($P^c(t)$) that opposes to it, resulting in a flattened $P(t)$. As mentioned in Section 4.1, Castillo-Cagigal (2014) developed a solution based on the first approach which consisted of building a distributed band-stop filter through the synchronization of the different users. The second approach is inspired in Active Noise Control (ANC) which is a method for reducing an unwanted sound by the addition of a second one specifically designed to cancel it. An example of this principle is described in Figure 4.7. The noise source is counteracted by another signal with the same amplitude, that will either phase shift or invert the polarity of the original signal. Then, both signals are combined into a new one. This process is called interference, and effectively cancels each other out (also known as destructive interference). In this Thesis, this approach is used to propose an adaptive algorithm in order to solve the problem in the time domain.

For a better explanation of this concept, imagine that there are only two facilities inside the grid: one non-controllable and one controllable. Both of them only consume power and there is no local generation. In order to simplify the environment and understand these concepts, imagine that both of them present a continuous and periodic consumption profile such as the one of the members of a real grid. Then, $P(t)$ as the sum of the two facilities would be periodic, being similar to the response of the aggregated consumption of a grid. The chosen signal to represent the consumption of the facilities is of sinusoidal nature being easy to use and describe its properties. Thus, $P^{nc}(t)$ and $P^{c}(t)$ have a sinusoidal waveform and they can be expressed as in Equation 4.6.

$$P^{nc}(t) = A^{nc} \cdot \cos(\omega t + \phi^{nc}) + \mu^{nc} \quad (4.6a) \qquad P^{c}(t) = A^{c} \cdot \cos(\omega t + \phi^{c}) + \mu^{c} \quad (4.6b)$$

where, A is the amplitude of the sinusoidal wave, ω is the frequency at which the sinusoidal wave is repeated, ϕ is the phase difference and μ is the mean value of the consumption since it is always greater or equal to zero. There exist two periodic consumptions greater than zero with the same periodicity and each period corresponds to a day. Varying the parameters of the sinusoid, different waveforms can be obtained which represent the consumption of each facility. An example of the waveform of $P^{nc}(t)$ and $P^{c}(t)$ is shown in Figure 4.8(a). The result of the aggregated consumption can be also observed in Figure 4.8(a). The form of $P(t)$ is also sinusoidal since it is the sum of two sinusoids of the same period and its expression is calculated in Equation 4.7.

$$\begin{aligned} &P(t) = P^{nc}(t) + P^{c}(t) = A_c \cdot \cos(\omega t + \phi_{nc})\mu^{nc} + A_{nc} \cdot \cos(\omega t + \phi_c) + \mu^c = A \cdot \cos(\omega t + \phi) + \mu \\ &where, \quad A = \sqrt{(A^{nc} \cdot \cos(\phi^{nc}) + A^{c} \cdot \cos(\phi^{c}))^2 + (A^{nc} \cdot \sin(\phi^{nc}) + A^{c} \cdot \sin(\phi^{c}))^2} \\ &\qquad\qquad \phi = \arctan\left(\frac{A^{nc} \cdot \sin(\phi^{nc}) + A^{c} \cdot \sin(\phi^{c})}{A^{nc} \cdot \cos(\phi^{nc}) + A^{c} \cdot \cos(\phi^{c})}\right) \\ &\qquad\qquad \mu = \mu^{nc} + \mu^{c} \end{aligned} \tag{4.7}$$

The resulting $P(t)$ is expressed in terms of $P^{nc}(t)$ and $P^{c}(t)$ and it also corresponds to a sinusoidal waveform depending on the parameters of the $P^{nc}(t)$ and $P^{c}(t)$. The three signals $P(t)$, $P^{nc}(t)$ and $P^{c}(t)$ are formed by a continuous part (μ) and an alternating part (sinusoid). Then, the algorithm seeks to cancel the alternating part of $P(t)$. In order to cancel it, the parameters of the controllable consumption are modified to adapt to the non-controllable part and the conditions are gathered in Equation 4.8.

$$\begin{aligned} &P(t) = A \cdot \cancelto{0}{\cos(\omega t + \phi)} + \mu \Rightarrow \begin{cases} P(t) = \mu = \mu^{nc} + \mu^{c}; \\ A_{nc} \cdot \cos(\omega t + \phi_{nc}) + A_c \cdot \cos(\omega t + \phi_c) = 0; \end{cases} \\ &\qquad A_c \cdot \cos(\omega t + \phi_c) = -A_{nc} \cdot \cos(\omega t + \phi_{nc}); \Rightarrow \begin{cases} A_c = -A_{nc} \\ \phi_c = \phi_{nc} \end{cases} \end{aligned} \tag{4.8}$$

The algorithm will reconfigure the controllable signal to create a destructive interference with those conditions and the aggregated consumption obtained is smoothed. Figure 4.8(b) shows the result of modifying $P^{c}(t)$ in order to cancel the alternating part of the $P^{nc}(t)$ getting a constant $P(t)$. In this case, it can be observed that one signal is opposite to the other one and they are in antiphase or opposite phase with the same amplitude. When one signal grows, the other one decreases to compensate the growth. This effect causes $P(t)$ being at equilibrium and constant over time.

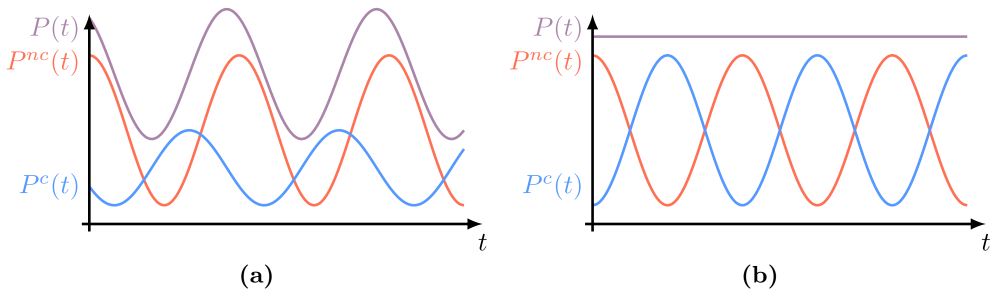


Figure 4.8: Environment simplification using sinusoidal signals as the demand profile for the different grid elements: (a) $P^c(t)$ is not adapted to $P^{nc}(t)$, so the form of $P(t)$ is not constant because the demand is arbitrary and (b) $P^c(t)$ is adapted to $P^{nc}(t)$, being $P(t)$ constant. $P^c(t)$ is drawn in blue, $P^{nc}(t)$ is drawn in red and $P(t)$ is drawn in purple.

In this case, the non-controllable demand has an analytical expression and it behaves the same over time. Thus, the construction of the controllable demand signal can be easily adjusted to interfere with the rest of the demand and be opposite to it. However, in general, there is no information about the analytical expression of $P(t)$ and how it evolves over time. So, the algorithm must ensure that the condition of Equation 4.4 is satisfied regardless the waveform of $P^{nc}(t)$. The algorithm will seek for a $P^c(t)$ that satisfies,

$$P^c(t) = C - P^{nc}(t) \tag{4.9}$$

Nevertheless, it is difficult to calculate the exact value of $P^c(t)$ needed to counteract $P^{nc}(t)$. The reason is that the information of $P(t)$ is not available at the moment the algorithm has to calculate and take a decision of what power the controllable facility has to consume. This measure is only available after all the components of the grid have consumed power and the grid response evolves depending on how much power was consumed. So, the algorithm has to apply a different strategy that is not based on the instant value of $P(t)$ but based on historical values enclosing information about it. A first approach should consist of using the tendency of $P(t)$. Thus, the condition, extracted from Equation 4.4 and expressed in Equation 4.5, is based on the trend of $P(t)$ and how it changes through time. If Equation 4.5 is expressed in terms of the components of the grid, the following relationship is achieved

$$\frac{dP(t)}{dt} = 0; \implies \frac{dP^c(t)}{dt} = -\frac{dP^{nc}(t)}{dt} \tag{4.10}$$

$P^c(t)$ in order to neutralise the variability of $P^{nc}(t)$ should grow when $P^{nc}(t)$ decays and vice versa. So the tendency of $P^c(t)$ should oppose to $P^{nc}(t)$ and it has to grow or decay at the same rate as $P^{nc}(t)$. However, there is no information about the form of the non-controllable demand. The only information available is $P(t)$ and the local power consumed by the facility $p_i^c(t)$. From the point of view of one controllable user, the rest of the demand is non-controllable. So suppose that for the jth controllable facility, the aggregated consumption from its point of view is as follows

$$p_j^c(t) = P(t) - P^{nc}(t) - \sum_{i=1}^{n-j} p_i^c(t) = P(t) - \hat{P}^{nc}(t) \tag{4.11}$$

where, $\hat{P}^{nc}(t)$ group together the rest of the grid consumption except the one coming from the local facility. Then, the local consumption of one facility is negligible compared to the sum of the rest of consumptions. Normally, a grid is composed by a huge number of facilities so it is normal that $\hat{P}^{nc}(t) \approx P(t)$. Thus, a controllable facility seeks to modify its consumption opposing to the rest of the components of the grid. If Equation 4.10 is developed for one individual, it can be obtained that

$$\frac{dp_j^c(t)}{dt} = -\frac{d\hat{P}^{nc}(t)}{dt} \approx -\frac{dP(t)}{dt} \tag{4.12}$$

In order to reduce the variability of the consumption, one controllable facility has a trend which is approximately the opposite to the aggregated consumption. Hence, $\hat{P}^{nc}(t) \approx P(t)$ when the number of individuals of the grid is high. Following the trend of $P(t)$ is the first step to counteract its variability. However, a problem arises when the n controllable facilities consume more than the m non-controllable facilities since $P(t)$ goes out of the constant equilibrium and a new disturbance has been created. Thus, it is necessary to predict the value of $P(t)$ and adapt to it in order to coordinate the different controllable facilities. That is why the use of ANN is appropriate and fits perfectly in the development of the proposed algorithm in this Thesis.

As explained in Chapter 3, ANNs have great capacities to act in distributed environments. They also stand out in the prediction of temporal series and their application to signal processing problems. In this Thesis, ANNs are used as the algorithm to manage the different consumptions inside a grid of the characteristics presented. Therefore, each facility will use a neural controller that has acquired the dynamic behavior of the grid and how to counteract the effects of the rest of the users based on the derivate of the aggregated consumption. ANN will know how to oppose to the derivate of $P(t)$ with the only information of the grid signal, $P(t)$, and the local behavior of the facility, $p_i^c(t)$. Once the ANN is trained, it will predict the next instant value of $P(t)$ and actuate to flatten it.

However, the communication in the environment is restricted and the information available is little. Another challenge to face is how the controllable facilities are coordinated to smooth the curve. The main reason why they have to act in a coordinated way is because otherwise once the equilibrium is reached, the system could be destabilized and $P(t)$ will not be flatten. So, the different neural controllers will self-organize to oppose to the non-controllable demand and stay at the equilibrium. For the rest of the Chapter, it is explained how to build an ANN able to compute the derivative of signal and oppose to an external stimulus. In this case, a restricted environment is used and it is comprised by only one controllable and one non-controllable facility to solve the first part of the problem. Whereas the collective part of the algorithm, it will be explained in Chapter 5.

4.3 Neural controller

In this Thesis, the use of ANNs is proposed to build a controller in order to modify the local consumption of part of the grid components will produce a smoothed aggregated consumption. The only information available is the history of the local consumption, $p_i^c(t)$, and the history of the aggregated consumption, $P(t)$. To produce a flattened $P(t)$, the idea consists of nullifying its derivative by opposing the amplitude to the non-controllable consumption. Hence, it is necessary to extract from the data the important information and elaborate a response to achieve a constant $P(t)$.

Because of the periodic nature of $P(t)$, an ANN could be applied to do a temporal series analysis and extract the tendency of $P(t)$ to predict its future values and adapt the response of the facility to achieve a flattened signal. There exist numerous architectures of ANNs as it is explained in Chapter 3. However, not all of them are fitted to solve the problem proposed in this Thesis. Thus, it is necessary a versatile ANN structure that is able to respond quickly to changes in the environment and adapt to them. There exist a number of structures that possesses these characteristics, being able to model the dynamics of the environment. RNNs fit perfectly for the application.

RNNs are commonly used in digital processing applications, but in this case the reasons to use them are: i) feature extraction in temporal series, ii) forecast

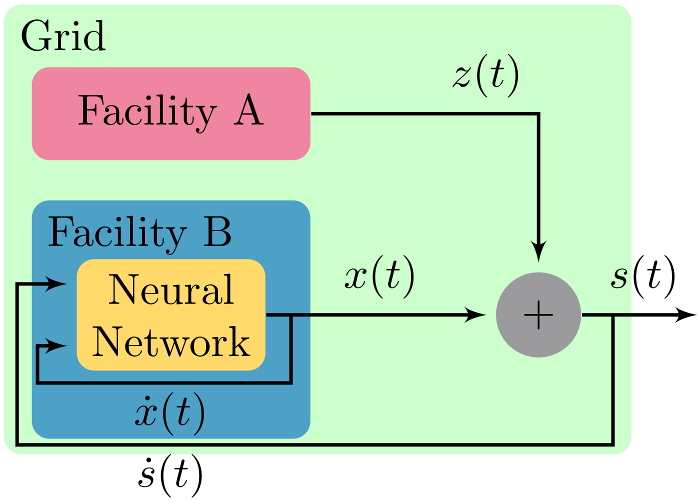


Figure 4.9: Electrical grid consisting of two facilities: Facility A does not have a controller and Facility B has a neural controller that control its demand. $z(t)$ represent the non-controllable demand, $x(t)$ represents the controllable demand and $s(t)$ is the environment signal or the aggregated consumption.

capabilities and iii) adaptive behavior in fast changing environments. In this way, the proposed RNN structure would be able to extract the periodic information to design a strategy in order to invert the polarity of the input the signal, with the peculiarity of not knowing the amplitude of the input signal at the moment of cancelling it. Therefore, the designed network is able to anticipate the values of the periodic input signal by extracting features of past input values. Then with that information, the RNN generates a response that it is in antiphase to the input at the same time step.

There are also different types of RNNs. In this Thesis, managing the consumption of a facility is done in real time, so it is necessary in some way that the RNN has any information about the time inside its structure. So, there are two possibilities to implement the neural controller: i) Discrete Time RNN (DTRNN) and ii) Continuous Time RNN (CTRNN). Both contain time information within their structure. However, the first one uses the discrete time to model a difference equation. While the second one, it uses continuous time to model a differential equation. As the application to the electrical grid is in continuous time, CTRNN are used as the structure to develop our neural controller. In addition, as described in Section 4.2, the ANN will be trained to follow the opposite of the input signal derivative. Thus, the network will have to compute internally that operation and respond to the input signal by modifying the environment, making necessary that the dynamic of the network also computes a differential equation.

In addition, the complexity of the problem makes difficult to design a CTRNN structure that solves the problem in only one step. Also, the grid is a very complex environment with many elements to take into account. Thus, a simplification of the grid has been done by dividing it into different parts to solve each one easily. The strategy of dividing and conquering will help to develop the neural controller that it is the fundamental block of the controllable facilities. Hence, different simplifications of the environment are made in this Part of the Thesis and the difficulty over its development will be increased.

To begin with, the reduced environment consists of only two facilities, one non-controllable and one controllable. Figure 4.9 shows the simplified environment with the two facilities. The grid has been divided in Facility A, which is the non-controllable part of the demand, and Facility B, which is the controllable one. Both facilities only consume power, there is no PV generation and no energy storage system. So, the only controllable element is the demand of the Facility B to be in antiphase to Facility A demand. The different signals of the environment are renamed in order to facilitate its nomenclature. $P^{nc}(t)$ becomes $z(t)$, $P^c(t)$ is $x(t)$ and $P(t)$ is renamed as $s(t)$. Then the environment equation is as follows,

$$s(t) = z(t) + x(t) \tag{4.13}$$

In Figure 4.9, it can be observed the information that the neural controller used to elaborate a response in antiphase to $z(t)$. Therefore, the controller has to extract the information from the history of the signals and build a destructive interference to obtain a flattened aggregated consumption. The inputs available are the derivatives of: the facility local behavior, $\dot{x}(t)$, and the environment aggregated consumption, $\dot{s}(t)$, both of them with respect to time. In this case the output of the neural controller is of the form,

$$x(t) = f(\dot{s}(t), \dot{x}(t)) = s(t) - z(t) \tag{4.14}$$

and the first step consists of building a dynamic model that emulates the antiphase behavior to the grid signal.

4.3.1 Neural structure

So far the grid has been simplified in two facilities in order to design the fundamental block of each controllable facility. The idea is to build a fundamental neural block that can be used by all the controllable facilities. However, there is not an exact formula to build the specific structure of an ANN for a general application. Normally, the trial an error method is used to discover the best structure suited for the application. In spite of this inconvenient, some tips have been followed, developed by the experience of working with ANNs. These tips are as follows:

- *Type.* The election of a type of ANN is closely related with the application itself. As explained in Section 4.3, it is required that the ANN has the ability to model a dynamic system in continuous time. Thus, the ANN is able to adapt to a fluctuating signal over the time and generate an output in antiphase to the environment signal. For this reason, CTRNN has been selected over the rest of types. CTRNN consists of a system of differential equations that models the dynamics of biological neural networks. Thus, CTRNN perfectly fits for the task at hand.

- *Layers.* The number of layers depend on the processing requirements of the network. If the problem needs that more features are extracted from the data, the number of layers will be higher. In addition, the dimensionality of the problem is also related with the number of layers, since more layers will provide the network with greater nonlinear capabilities. Thus, normally the architecture will have an input layer, where the data is processed and encoded to the neural space, and an output layer, where the network response is returned to the environment in which it is used. Moreover, one or more hidden layers can be used to increase the processing capacities and divide the problem into more neural dimensions. In this Thesis, the neural structure selected is composed by one input layer, one or two hidden layers, that will be explained later, and an output layer. The number of neurons is as follows.

  - *Input.* This layer is composed by only two neurons because there are only two inputs, $\dot{s}(t)$ and $\dot{x}(t)$. The idea is that both neurons receive both inputs multiplied by a gain. Thus, those neurons will process the inputs and extract some features from it.
  - *Hidden.* There is no clear reason for choosing the number of hidden layers. In this Thesis, the number of hidden layers is selected between one or two layers since more hidden layers will complex too much the problem and the processing speed will decrease as more neurons are added. The number of neurons in each layer is also not clear, they are varied from 2 to 4 neurons depending on speed, complexity and processing capacities of the

layer. Hence, the best size and number of hidden layers are determined for the proposed problem.

  – *Output*. There are only one output, so the number of neurons in the output layer is one. In this case the output of the network is directly $x(t)$ which is the signal in antiphase to cancel the fluctuations of the environment signal.

- *Neuron function*. The function that a neural network computes is extremely related to the application in which is used. In this case, our application consists of canceling the derivative of $s(t)$ with respect to time, so it is necessary a network that incorporates a dynamic behavior plus a differential equation formulation. Those are the main reasons to use CTRNN. The operation of a CTRNN neuron is a differential equation that models the behavior of a biological neuron. Recalling Chapter 3, the function that a CTRNN neuron computes is described in Equation 4.15.

$$\dot{y}_i(t) = f_i(I_i(t), y_1(t), \ldots, y_n(t)) = \frac{1}{\tau_i} \cdot \left(-y_i(t) + \sum_{j=1}^{N_{pre}} w_{ij} \cdot \sigma_i\left(y_j(t) + \theta_j\right) + \sum_{m=1}^{n_{in}} g_m \cdot I_m(t) \right)$$
$$with\ \sigma_i(u) = \frac{1}{1+e^{-u}} \tag{4.15}$$

  where, y_i is the activation of the ith neuron, $\dot{y}_i$ is the rate of change of the ith activation neuron, τ_i is the time constant, w_{ij} is the connection weight of the jth to the ith neuron, $\sigma_i(\cdot)$ is the activation function, θ_i is the bias of the neuron and I_m is the external input of the neuron (if any), which is multiplied by a gain, g_m. In Equation 4.15 there are some changes compared to Equation 3.8 because neurons of the input layer have more than one input and are multiplied by a gain. This gain is used to prepare the inputs to enter in the network. The network is composed by homogeneous neurons, so all of them compute the same function. The output of each neuron of the network is equal to $\sigma_i\left(y_i(t) + \theta_i\right)$. Thus, the controllable demand would be equal to the output of the last neuron, i.e. $x(t) = \sigma_n\left(y_n(t) + \theta_n\right)$.

- *Flow of information*. The information inside the network can go forwards and/or backwards, depending on the connection among the neurons. It is important to establish a hierarchy in case that more than one layer is used and if a dynamic behavior is to be achieved, feedback loops are also required to be inside the structure. Thus, different connections are established between layers to form a hierarchical structure in which the processed information goes forward from one layer to the other one. The information goes from the input layer to all the neurons of the hidden layer. In the hidden layer the information is processed in all the neurons and goes to next layers. Finally, it reaches the output layer. Moreover, the information also goes backwards due to the presence of feedback loops. Each neuron of the input layer has a feedback loop from its output to its input. In the hidden layer the feedback loop is more complex, the output of the neurons are connected from the output to the input of the rest of the neurons and itself. The structure of the hidden layer resembles to the one of the Elman networks (Elman, 1990) in which the state of the network is stored in a group of neurons connected to the hidden layer. Finally, in the output layer, the neurons are connected as in the input layer.

Based on these tips, a first approximation of the neural controller structure is elaborated. Figure 4.10 shows this structure and the different synaptic connections between the neurons. There are as many input neurons as input signals available, i.e. two units, and there are as many output neurons as outputs required, i.e. one unit. However, there is not a specific way to determine the number of processing units in the hidden layers and how they are connected (1 or more hidden layers). Section 4.3.2 consists of explaining the training performed to obtain the structure of Figure 4.10 that solves the problem at hand.

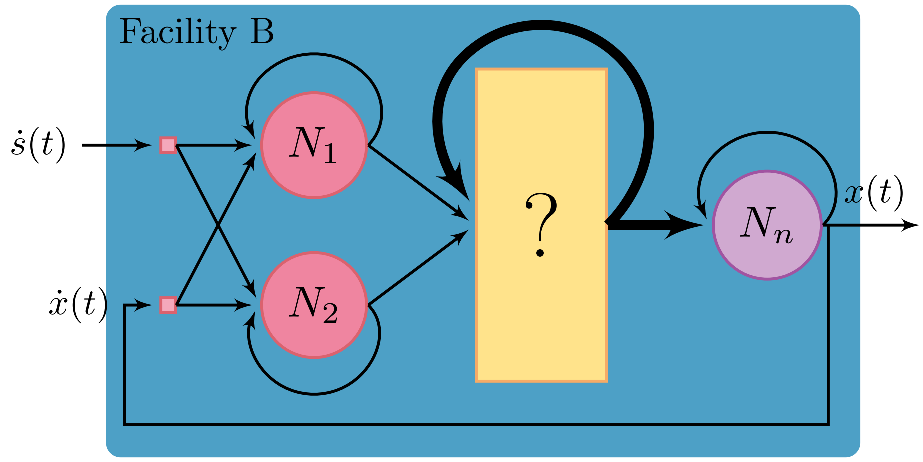


Figure 4.10: Structure of the neural controller for the controllable facilities. The ? block corresponds to the structure of the hidden layer that is not decided yet.

4.3.2 Neural Parameters

A first neural approximation was defined for the controller that is inside the controllable facility. However, there are some aspects not clear in its definition, which are the hidden structure of the controller and how to train the network in order to fulfill the task of canceling the environment signal. In order to decide the final neural architecture, different simulations have been prepared in which the performance of various ANN architectures are compared to obtain the best one. The only way to evaluate their performance consists of comparing if the different neural architectures can solve the problem of neutralising the fluctuations of the demand. Thus, the training algorithm and the election of the different variables to adjust it, are also important in order to get the objective previously mentioned.

A Genetic Algorithm (GA) has been used to adjust the free parameters of the neural structure. The reasons are that everything necessary is known: i) the behavior that the network has to adopt and ii) the function that it may compute to achieve a flattened environment signal (see Section 4.2). The definition of a fitness function could be based in Equation 4.12, evaluating the result of applying the neural controller. However, it is also necessary to decide the different parameters of the CTRNN and select the best configuration of the GA to evolve successfully the network. With the tuning, a neural structure is achieved in order to predict the next value of the input signal coming from the environment by being able to compute the opposed derivative of $s(t)$. It is also required that once the network is tuned, its output could adapt to changes in the waveform of $s(t)$ because its response affects the environment. Ensuring a proper generalization of the CTRNN controller makes possible to use it in different provided environments. These environments have different $s(t)$ of similar nature (continuous, periodic and of class C^1) to the one used during the evolution.

The main objective of the GA consists of evolving the neural structure of the CTRNN in order to minimize the fitness function define to this problem. However, the structure of the CTRNN is not completely decided, so there is more than one structure to tune. Thus, all the possible neural parameters are divided into two groups:

- *Structures.* 12 structures are going to be evaluated to select the best combination of neurons that obtain the best performance in the generation of an antiphase signal. These 12 structures use different number of neurons distributed in one or two layers depending on the structure. A higher number of layers would make the network too complex, being slower to train. Also, the increased processing capabilities of more layers would not improve the

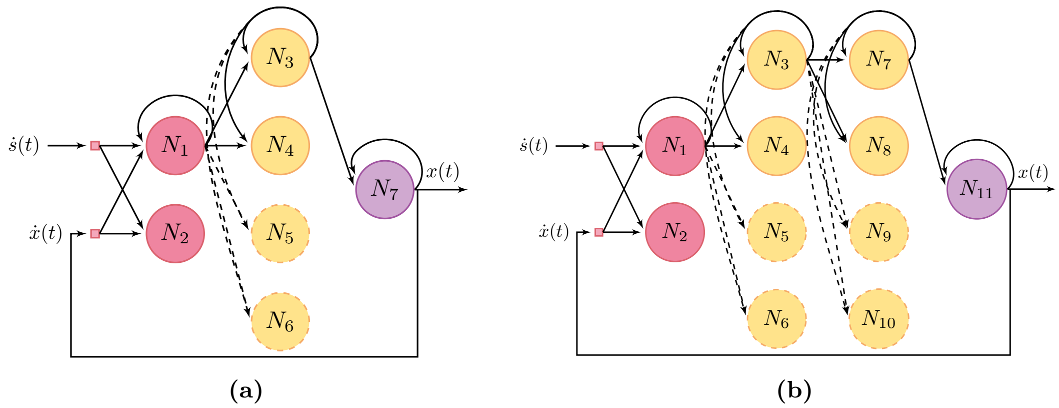


Figure 4.11: CTRNN structure of the hidden layer: (a) one hidden layer and (b) two hidden layers. The synaptic weights of each neuron of the layer is represented in the first neuron of it. The minimum size of the hidden layer is two neuron (drawn with solid lines), whereas the maximum number of neurons is four (drawn with dashed lines).

performance of the network since the available environment information has only one component. In Figure 4.11, it is condensed the different configurations of the hidden layer. Figure 4.11(a) represents the architectures with one hidden layer. Whereas Figure 4.11(b) shows the architectures with two hidden layers. In each layer of the structure hidden part, the number of neurons varies from 2 (solid units) to 4 neurons (dashed units). Making a total of 3 different architectures with only one hidden layer and 9 architectures with all the possible combinations of neurons in both layers. 2 neurons were chosen as the minimum number of neurons because at least the network will need the same units as the input layer. On the contrary, 4 neurons were chosen as the maximum number of neurons because it is the double of the units in the input layer and it will be enough to process the incoming information of the previous layers.

- *Free parameters to adjust.* From Equation 4.15, the free parameters of the network are: τ_i, w_{ij}, θ_i and g_m. g_m and w_{ij} can be grouped together because they represent the connections of different parts of the network. The value of τ_i is going to be fixed to 1 because it is not necessary to variate the reaction speed of the neurons at the moment. So the parameters of the network used during the evolution are w_{ij} and θ_i. However, it is unknown the numerical range in which these parameters are optimum and solve the problem. Thus, 6 different intervals are used in order to know in which range of parameters the network performs better and achieves the best results. These 6 intervals are divided in two types: $[0, a]$ and $[-a, a]$ with $a = 1, 5, 10$. Thus, there are 3 intervals with only positive values and 3 with positive and negative values. The reason is that it is necessary to test if the problem can be solved with only excitatory connections (positive values) or inhibitory weights (negative values) must be added. The 3 different values of the interval extremes are decided in order to know how large might be the parameters to refine the search and obtain the best combination of parameters. A small interval ($a = 1$), a medium interval ($a = 5$) and a large interval ($a = 10$) are used. Both the synaptic connections, w_{ij} and g_m, and the bias, θ_i, take their values from the same interval. In each simulation scenario, only one parameter interval is used generating 72 experiments for each structure and interval.

In addition, for the simulations and training of the network a computer was used. In order to implement the CTRNN inside a computer program, it was necessary to discretize the differential Equation 4.15. Recalling the definition of the derivative of a function $f(t)$ with respect to t,

$$\frac{df(t)}{dt} = \lim_{\Delta t \to 0} \frac{f(t+\Delta t) - f(t)}{\Delta t} \tag{4.16}$$

However, in a discrete machine this Δt can be as small as wanted but not 0, so it is going to be approximated by the division of finite intervals. If this approximation is used with the differential term of Equation 4.15, it will be obtained that

$$\frac{dy_i(t)}{dt} \approx \frac{y_i(t+\Delta t) - y_i(t)}{\Delta t} \tag{4.17}$$

Then, by substituting the result of Equation 4.17 into Equation 4.15 and using the discrete time nomenclature consisting of changing t by k, Equation 4.18 represents the operation of the CTRNN.

$$\begin{gathered} y_i[k+\Delta k] = y_i[k] + \tfrac{\Delta k}{\tau_i} \cdot \Delta y_i[k] = \\ = y_i[k] + \tfrac{\Delta k}{\tau_i} \cdot \left(-y_i[k] + \sum_{j=1}^{N_{pre}} w_{ij} \cdot \sigma_i\left(y_j[k] + \theta_j\right) + \sum_{m=1}^{n_{in}} g_m \cdot I_m[k] \right) = \\ = \tfrac{1}{\tau_i} \cdot \left((\tau_i - \Delta k) \cdot y_i[k] + \Delta k \sum_{j=1}^{N_{pre}} w_{ij} \cdot \sigma_i\left(y_j[k] + \theta_j\right) + \Delta k \sum_{m=1}^{n_{in}} g_m \cdot I_m[k] \right) \end{gathered} \tag{4.18}$$

Finally, another free parameter appears, the time interval of the derivative. In this case, only one sample ($\Delta k = 1$) is used so the response of the next step will depend only on the previous response of the network.

4.3.3 Genetic Algorithm configuration

In this Thesis, a basic GA is used to evolve the CTRNN structure. This GA is composed by the three basic operations, selection, crossover and mutation, over a population of individuals whose performance will be evaluated based on a $FF(\cdot)$. So far the different parameters to be evolved were introduced for the different neural structures. However, it is necessary to configure the different parameters of the GA in order to reach a solution and evolve correctly. The different parameters of the GA and its values are as follows:

Population

A population is composed by a number of individuals, which each one contains a solution to the problem. Then, the performance of each of them is evaluated by the $FF(\cdot)$ to find the best solution to the problem. Each individual of the population contains a *chromosome* with the set of parameters to evolve. The *chromosome* contains the different parameters to adjust from the neural controller and its values are between 0 and 1. Then, depending on the interval selected for each parameter, each value of the chromosome is adjusted to evaluate the performance of the solution. Each generation, a new population is created until it reaches the limit of generations or a plateau value. The new population is formed based on the best individuals of previous generations. Thus, it is necessary to configure different parameters related with the population to reach the best solution. These parameters are as follows:

- *Chromosome*. It contains the different values to adjust, so that the length of the chromosome depends directly from the architecture of the CTRNN. A relationship can be established between the chromosome and the parameters of the neural network. Thus, the chromosome is composed by the different w_{ij} and θ_i, which is expressed mathematically in Equation 4.19.

$$l_{ch} = \overbrace{N_{in} \cdot (n_{in} + 1)}^{input} + \overbrace{\sum_{l=1}^{L} N_{h,l} \cdot (N_{pl,l} + N_{h,l})}^{hidden} + \overbrace{N_o \cdot (N_{pl} + 1)}^{output} + \overbrace{N_T}^{bias} \quad (4.19)$$

where, l_{ch} is the length of the chromosome, N_{in} is the number of neurons in the input layer, n_{in} is the number of inputs associated with the g_m terms, $N_{h,l}$ is the number of neurons of the lth hidden layer, $N_{pl,l}$ is the number of neurons of the previous layer, N_o is the number of neurons in the output layer and N_T is the total number of neurons. Over each term of Equation 4.19 is the name of the part of the neural structure to which corresponds. The minimum length of the chromosome corresponds to the neural architecture with only 1 hidden layer and 2 neurons in it, and it has a value of 22 genes. Whereas the maximum length corresponds to the structure with 2 hidden layers and 4 neurons in each of them, that is 78 genes. The length of the rest of structures is between 22 and 78 genes, so it is also necessary to adjust correctly the parameters of the GA to find the best solution in each case.

- *Size.* The population size affects the speed to find the solution. The higher the size of the population is, the greater the number of solutions are tried. So, there are more possibilities to find the best solution for big populations. During the tuning process, different population sizes are tested to find the optimum number of individuals per generation. The size of the populations varies between 20 to 100 individuals with intervals of 10 individuals of difference between them. Thus, 9 different populations sizes are tested.

- *Generations.* It is also required that evolution is limited in order to finish running the GA. In first place, the number of generations should be high enough so that the algorithm can reach the solution within it. On the other hand, this number guarantees that the simulation stops in a hard limit. In case that it is a large number, it would take a long time for simulating it when any optimal solution is not reached. For these simulations, this number was fixed to 1000 generations because it is enough to prove the validity of the CTRNN under the parameters found.

Selection

This is one of the genetic operators that builds the next generation of individuals. It is based on the mechanism described in Section 3.5.2.2. The one selected is the roulette-wheel method. This operator consists of reordering the individuals based on their fitness value. The bigger the fitness value is, the bigger chances the individual has to be selected. Then, a number of individuals are selected randomly based on their probability. The number of selections is based on the population size. However, based only on the randomness of the process, diversity and the best fitted solution of the problem can be lost. To solve this problem, a combination of the roulette-wheel method with the elitism is selected. Thus, a number of individuals of the next generation is reserved to copy directly which consists of the best individuals of the previous generation. In this case, the number of elites is fixed to 6, being enough to breed the next generation.

Crossover

The crossover operator is applied over the individuals obtained from the previous selection. This operator consists of creating a new offspring from the selected individuals of the previous generation that would breed new individuals in the next generation. The idea of the operator is to choose two individuals, select randomly a point in their chromosomes and then swap the parts of the chromosomes. Therefore, two individuals are created for the new generation with part of their predecessors. The crossover favors the exploration of the search scenario. There are different crossover operators, but the simplest one was used to make the GA quick and to prove that the network can be trained with the simplest algorithm. The crossover operator only selects 1 point to cross the chromosome. This point is randomly chosen among the different genes of the chromosome.

Mutation

This is the last genetic operator implemented. It consists of changing the value of a gene inside the chromosome randomly with a very low probability. In this case the value of the gene is altered following a Normal Gaussian distribution with a variance of $\sigma = 0.2$, whose value will be added to the gene in order to modify its value. This operator is used due to the introduction of diversity inside the population. It also helps to avoid favoring one individual against others in the genetic drift, to avoid the premature convergence of the algorithm and to guarantee that the algorithm will found the global solution with enough time. Different mutation rates are used in order to find the best one for the problem. These rates are between 0.01 and 0.1 in steps of 0.01. So, there are 10 different mutation rates to select the best choice for our algorithm.

Fitness function

The last element of the GA configuration is the evaluation of the individuals. $FF(\cdot)$ is in charge of evaluating the performance of the different chromosomes once they have been used for its purpose inside the simulation as the parameters of the CTRNN. The form of this function is defined by the user, but it should contain the elements needed to minimize the objective of the neural structure selected. In this case, a function is built to be capable of evaluating the capacity of the corresponding CTRNN structure to be in antiphase to the C^1 input signal. In Section 4.2, it was concluded that a controllable user has to behave as Equation 4.12 to get an antiphase signal. Hence, based on Equation 4.12, a first approximation of the $FF(\cdot)$ is as follows (see Figure 4.12(a)).

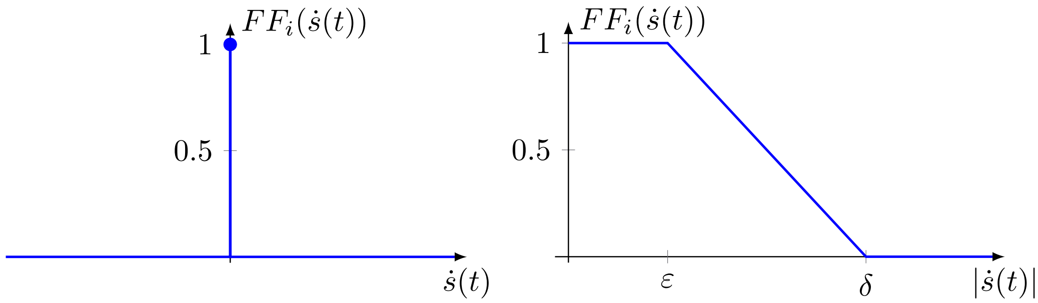
$$FF_i\left(\dot{s}(t)\right) = \begin{cases} 1; & if\ \dot{s}(t) = 0 \\ 0; & otherwise \end{cases} \tag{4.20}$$

where $FF_i\left(\dot{s}(t)\right)$ is the fitness value of an individual. However, a GA does not behave well with discontinuous fitness functions like the one formulated in Equation 4.20. The reason is that it seems like only the individuals that reach the condition have a fitness value, the rest are equal to 0. So, imagine that neither of the individuals get a value different from 0, the GA could not evolve because none of the solutions has a fitness value. That is to say, a GA with such a $FF(\cdot)$ would not converge to an appropriate solution to the problem.

Thus, it is necessary to change the abrupt formulation of Equation 4.20 in order to make the algorithm converge to the solution. To do that, the condition of the $FF_i\left(\dot{s}(t)\right)$ is relaxed and a gradual transition is made between the two states (see Figure 4.12(b)). With the relaxation of the condition, more individuals can score and evolve to better solutions in order to reach the maximum fitness. Equation 4.21 shows a gradual fitness function formulation of Equation 4.20.

$$FF_i\left(\dot{s}(t)\right) = \begin{cases} 1; & if\ |\dot{s}(t)| < \varepsilon \\ -\dfrac{|\dot{s}(t)| - \delta}{\delta - \varepsilon}; & if\ \varepsilon \leq |\dot{s}(t)| \leq \delta \\ 0; & if\ |\dot{s}(t)| > \delta \end{cases} \tag{4.21}$$

where, ε is the inferior limit and δ is the superior limit. Being $\varepsilon << \delta$. In Equation 4.21, the direction of the derivative is not taken into account because the algorithm



(a) **(b)**

Figure 4.12: Fitness function representation to evaluate the derivative of the environment signal: (a) only when the derivative is 0 the individual scores and (b) gradual change from maximum to minimum.

seeks for the reduction of its slope. Figure 4.12(b) shows a graphical representation of this $FF(\cdot)$, in which the maximum and minimum value are linked by a ramp. Thus, more individuals can score and the GA can decide between multiple options to evolve the result. ε and δ have been chosen carefully. If ε is too low, the solution is nearer to $\dot{s}(t) = 0$ and the GA will take much more time to reach it. If $\varepsilon \approx \delta$, the fitness function have a discontinuity of two states, being difficult to evolve. If δ is too high, then almost all the individuals were able to score and the GA would not be able to evolve. Based on these concepts, these limits are fixed to $\varepsilon = 10^{-5}$ and $\delta = 10^{-2}$.During the simulation results, this assumption will be checked.

In addition, as it is explained for the CTRNN discretization, the GA is going to be used inside a computer to train the network. Then, it is necessary to discretize the $\dot{s}(t)$ used in the fitness function. Thus, a first approximation of $\dot{s}(t)$ is based on the definition of the derivative.

$$\dot{s}(t) = \lim_{x \to 0} \frac{s(t + \Delta t) - s(t)}{\Delta t} \approx \frac{s[k + \Delta k] - s[k]}{\Delta k} \underset{\Delta k = 1}{=} s[k+1] - s[k] = \Delta s[k] \quad (4.22)$$

However, from the difference of one sample, it might be possible that the CTRNN could not extract the tendency behavior of the signal. Thus, a second definition is proposed consisting of the mean of an interval of samples of the environment signal. Equation 4.23 reflects this behavior.

$$\Delta s[k] = \frac{1}{N_s} \sum_{l=k-N_s}^{k-1} s[l+1] - s[l] \quad (4.23)$$

where, N_s is the number of samples over with the mean is elaborated and with $k \geq N_s$. In this case, a value of $N_s = 10$ samples is used to verify that this approximation is correct and the better results than the instantaneous difference are obtained.

So, the same $FF_i(\Delta s[k])$ function has been defined but with 2 approximations of the derivative of the environment signal. Equation 4.21 shows the value of fitness for only one instant of time, so for the whole time length the value of individual fitness is in Equation 4.24.

$$FF_i\left(s\right) = \frac{1}{K_s} \sum_{k=0}^{K_s - 1} FF_i\left(\Delta s[k]\right) \quad (4.24)$$

where, K_s is the total number of samples of $s[k]$. The fitness values of the different time instants have been integrated and then make the mean of all of them to obtain the value of an individual. Consequently, the average fitness value of the population is computed in order to know how well the performance of the generation was during that realization (see Equation 4.25).

$$FF(s) = \frac{1}{N_{pop}} \sum_{i=1}^{N_{pop}} FF_i(s) = \frac{1}{N_{pop} \cdot K_s} \sum_{i=1}^{N_{pop}} \sum_{k=0}^{K_s-1} FF_i(\Delta s[k]) \tag{4.25}$$

where, N_{pop} is the number of individuals of the generation. Finally, each generation is evaluated over and over again until the last generation or a plateau value is reached.

To sum up, 2 fitness functions are evaluated based on how the derivative of the $s(t)$ is done (see Equations 4.22 and 4.23). And then, it will be decided which one is the best form to extract the derivative behavior of the signal.

Seeds

The last parameter of the algorithm is based on the starting point of the search to find the best solution of the problem. Normally, the field of search is initialized in different random points to avoid the problems of convergence in a finite time and to find a faster solution of the problem. In addition, beginning in different points of the search space will allow the GA to be independent of the initial conditions. For each experiment, 30 different seeds have been used to randomized the initialization of the algorithm. Thus, each experiment is repeated 30 times in different parts of the search space. With this number of seeds, it is enough to discover the dependency with the initial conditions and if the best solutions are reached for these configurations of the GA.

4.3.4 Stability

It is necessary to analyze the stability of these algorithms to determine if they converge. The stability of the CTRNN and the GA used to train it must be guaranteed. Thus, some tools are used to assure that they are stable.

In first place, the stability of the RNN was already contemplated in Section 3.3.1. In general, the stability analysis of RNN is complex, because each neuron represents a difference or differential equation in an N-dimensional space. However, there are some theorems based on the Lyapunov stability theory (Lyapunov, 1992) that establish a way to study the dynamic of the system represented by the CTRNN. Based on this theory, Cohen and Grossberg (Cohen and Grossberg, 1983) develop the Theorem 3 that will assure that a RNN is stable in the Lyapunov sense provided that $\boldsymbol{W}$ is symmetric, the y_i of the neuron is nonnegative and the $\sigma_i(\cdot)$ is monotonically increasing. If these three conditions are met, then the proposed CTRNN will satisfy the theorem and will be stable. By construction, the CTRNN has a $\sigma_i(\cdot)$ that is monotonically increasing since it is a sigmoid function. In addition, the function that each neuron computes is nonnegative since its output is directly the activation and the range of the sigmoid is between 0 and 1. Thus, two of the three imposed conditions are fulfilled. However, the final form of the matrix $\boldsymbol{W}$ is unknown and its information is encoded inside the chromosome of the GA. So, this condition will try to be satisfied in the process of finding the best solution to the problem.

The stability of solutions of the GA was already demonstrated in Holland (1975) and encouraged by Golberg (2006). They invented the *Schema Theorem* to prove the convergence and stability of the GA. The GA is composed by different schemas. A schema represents a group of individuals in the solution space, and corresponds to attributes of those individuals. The *Schema Theorem* reveals how a fitted schema will survive and multiply along the evolution, whereas the number of unfit individuals will decrease.

This theorem explains how new individuals are obtained and the growth ratio depends on the characteristics of the chromosome and the number of fixed strings inside it. So the algorithm will continuously search for better fitted individuals with a growth ratio depending on the population average fitness and the probability of

ANN Parameters	min	max
$\boldsymbol{W}$ and g_m	17	67
θ_i	5	11
τ_i	–	1
# hidden layers	1	2
# hidden neurons	2	4
Total # of neurons	5	11
Total # structures	12	
Total # intervals	6	

Table 4.1: Configuration of the ANN for the different simulations.

GA Parameters	min	max
chromosome length	22	78
Population size	20	100
# generations	–	1000
# elites	–	6
# crossovers	–	1
Mutation rate	0.01	0.1
# $FF(\cdot)$	–	2
# seeds	–	30

Table 4.2: Configuration of the GA for the different simulations.

surviving the operations. To sum up, both of the algorithms used (CTRNN and GA) are built following both theorems (Cohen-Grossberg Theorem and Schema Theorem) so that they provided the necessary conditions to assure the stability of the algorithms.

4.4 Simulation Results

At the end of the training for all the different simulation configurations, the best CTRNN architecture and the best parameters of the GA will be obtained. From Sections 4.3.2 and 4.3.3, the different parameters of the simulations are reduced to Tables 4.1 and 4.2. Thus, there are 12 different CTRNN structures which use 6 different intervals for their free parameters and they are trained with a GA that has 9 different sizes of populations evolved in 1000 generations with the 3 basic genetic operators: selection with 6 elites, 1 crossover and 10 different mutations rates, evaluated for 2 fitness functions. And each experiment is repeated 30 times to randomize the initial conditions of the algorithm. So the number of simulations to be done is 388800 in total.

All the simulations under the set of parameters of Tables 4.1 and 4.2 are conducted by *NeuralSim* simulator. *NeuralSim* is an open source simulator under GPLv3.0 license developed in C++. It was build as a general purpose simulator to prove different neural structures with different methods to train them, learning algorithms and GAs. The simulator is built following a modular structure so anyone can configure different neural structures with different neural functions. The different parameters of the neural network and the algorithm used to train it, are configured through different XML files. All the simulations were run in a computing cluster of 60 unit processors of *AMD Opteron(tm) Processor 4180* with 40 GB of RAM memory, under a distribution *Rocks 6.1.1 (Sand Boa)* of 64 bits architecture.

Before presenting the simulation results, it is needed to talked about the input signal of the neural controller. Its aim is to smooth the aggregated consumption of a grid, but a more general input signal is used for this preliminary study to determine the best configuration of the CTRNN and the GA. Thus, a continuous periodic signal of class C^1 is required to evaluate the neural structure and check if the derivative intuition to smooth a curve is correct. In this case, a sinusoidal function is used which possesses all characteristics and similar properties of the one to smooth. In addition, it is also necessary a representative waveform of the varying consumption with its peaks and valleys. Therefore, a sinusoid is used with a day period to simulate a peak and a valley. However, a high resolution sinusoid is not used with 1440 samples per period as it length in minutes of a day. The idea is to use a sampled sinusoidal of 72 samples per period so that it is the same as sampling the daily aggregated consumption every 20 minutes. Moreover, the input signal is formed of 3 periods of

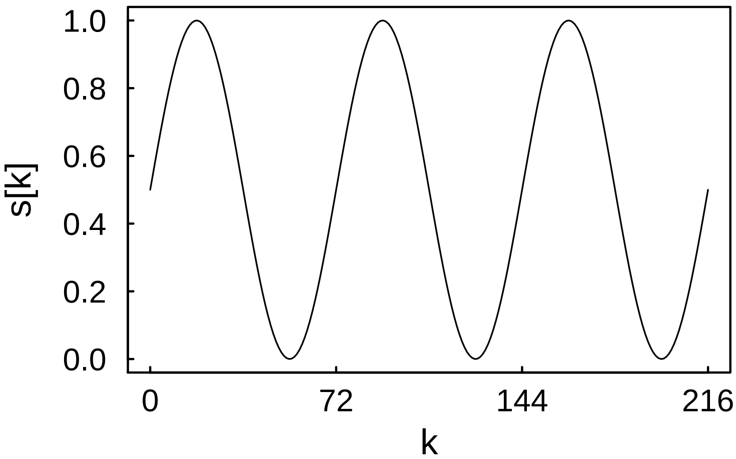


Figure 4.13: Input signal used for the training, consisting of a 3 periods sinusoidal function of 72 samples per period.

72 samples in order to isolate the initial state of the CTRNN and reach a stable state to evaluate the performance of the neural controller. The representation of the input signal is shown in Figure 4.13. As observed in Figure 4.13, the sinusoid presents enough information required by the CTRNN to compute the inverse derivative.

All the elements involved in the training are already presented as well as the tools used. Now, it is necessary to analyze the different results obtained after the 388800 simulations. The analysis consists of evaluating the performance of the different structures with the different configuration parameters. Thus, the analysis consists of going from the general, evaluating the overall assessment of each structure, to the specific, how this performance is affected by the different parameters simulated.

Neural structures and their parameters performance

Firstly, Figure 4.14 shows the overall performance per structure for all possible combination of parameters. The nomenclature followed for referring to each structure is of the form $\#ML\#N$, where $\#M \in \{1, 2\}$ is the number of hidden layers of the structure, and $\#N \in \{2, 3, 4\}$ is the number of neurons of the hidden layer, $\#N \in \{2, 3, 4\}$. At a first glance, the performance of each structure is very similar. All boxplots are composed by 32400 points corresponding to the individual with the best fitness value reached during the 1000 generations for all the different configurations of the CTRNN and GA parameters.

For the 12 structures, the value of the higher whisker is located closely to 1.0 which is the maximum value that the $FF(\cdot)$ can take. This assures that in all the structures, there is a combination of parameters that ensures that $\dot{s}[k] < 10^{-5}$, where 10^{-5} is the higher limit of the fitness function. On the other hand, the minimum fitness value for all the 12 structures is also near the vicinity of a 0.2 fitness value. Thus, in those cases, the environment signal variations are in between 10^{-2} and 10^{-5} (the limits of the fitness function) since the average fitness value is different from 0. Both whiskers are situated to 1.5 the value of the InterQuartile Range (IQR), or $1.5 \cdot |Q3 - Q1|$. The IQR is a measure of the data deviation, so in this case a great deviation is observed in the data of the simulations. Again for all the structures the $Q1$ is located above 0.2 but closes to it and the 25 % of the simulations reached this maximum fitness. Close to the $Q1$ is the median which represents the 50 % of the data. The median for all structures is close to a fitness value of 0.25, however structures with 4 neurons in the last hidden layer have a little lower median value.

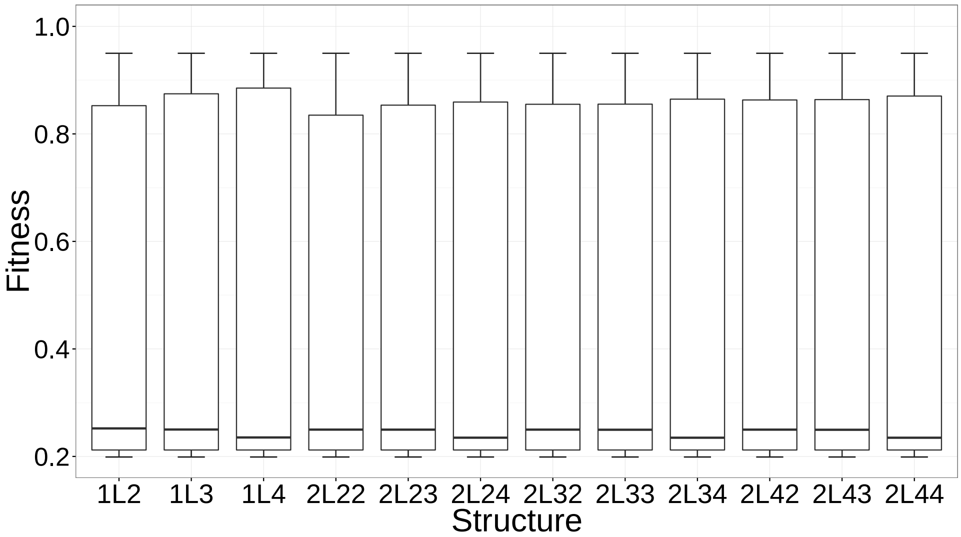
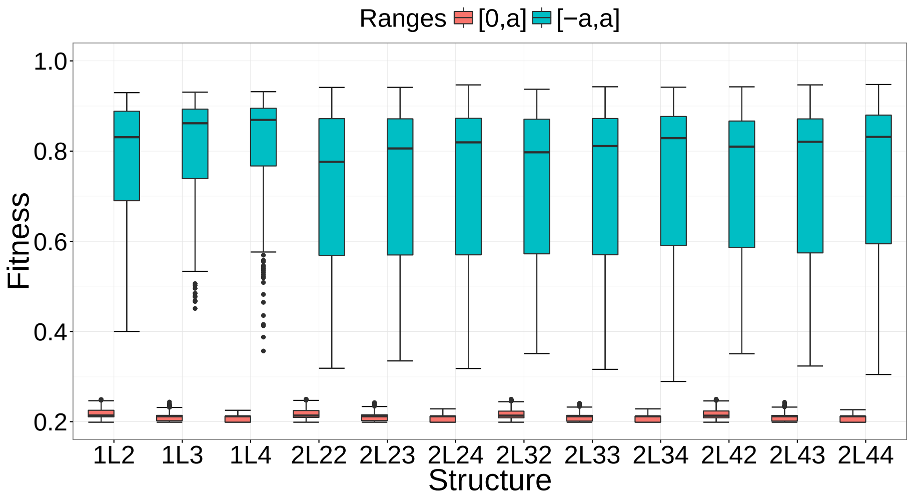


Figure 4.14: Overall fitness performance per structure for all the possible configurations of the simulations. The nomenclature of the structure is $\#ML\#N$ with $\#M$, the number of hidden layers, and $\#N$, the number of hidden neurons. Each box comprises observations ranging from the first to the third quartile. The median is indicated by a horizontal bar, dividing the box into the upper and lower part. The whiskers extend to the farthest data points that are within 1.5 times the interquartile range. Outliers are shown as dots.

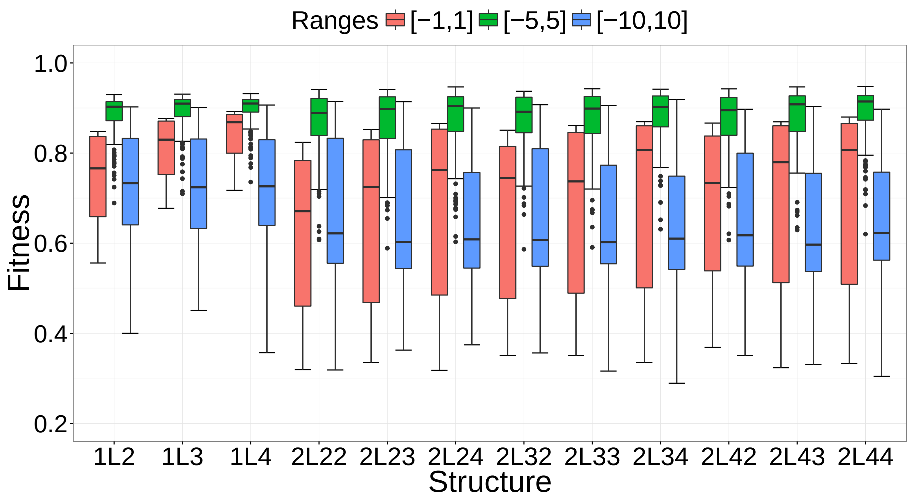
Finally, the $Q3$ for all the simulations is above a 0.8 fitness value, so that the 25 % of the simulations are above this value. Thus, for these 25 % of simulations, it is closer to the higher limit of the fitness function and the amplitude of the variations of $s[k]$ are almost reduced. In addition, it can be observed, some differences between the values of $Q3$ depending on the structure. The best value of the $Q3$ corresponds to the structure $1L4$ (1 hidden layer with 4 neurons), but in general the performance of the architectures with only one hidden layer are only 1 % above the two layers architectures. This figure seems relatively too low to state which structure is the best. However, it can be concluded that the complexity introduced by the two hidden layers do not achieve better results than the ones with a hidden layer. The tuning is reduced since fewer parameters are used and the processing capabilities are more fitted to requirements of the problem.

Nevertheless, the information of Figure 4.14 is not enough to reach any conclusion. So, the data corresponding to each structure are split in the different ranges of the tuple $\{\boldsymbol{W}, \boldsymbol{\Theta}\}$. Figure 4.15 shows all the results for the different ranges of values that the parameters of the CTRNN can take. In order to reduce the volume of data, the seeds of the different configurations have been grouped to analyze the mean performance of each evolution since all the structures behave similarly. A first division has been made based on the form of the value ranges. The two ranges selected are: i) $[0, a]$, only positive values, and ii) $[-a, a]$, symmetric range of values. The first range indicates that there is no inhibition inside the neurons, so the connections and bias will excite the next response coming from the previous neurons. On the contrary, the range of values of the form $[-a, a]$ allow neurons to have excitatory or inhibitory connections which increase the adaptivity and the regulation of the neural activation. These assumptions are also corroborated by the results shown in Figure 4.15(a).

For all the structures, when $\{\boldsymbol{W}, \boldsymbol{\Theta}\} \in [0, a]$ there is no difference in the fitness value for each one. In all these cases, the structure is far from reaching the maximum value of the fitness, so hardly these structures are evolved. This behavior occurs due to the absence of inhibitory synapses. The neurons are saturated due to the



(a)



(b)

Figure 4.15: Performance of the 12 structures divided by the range of values that $\{\boldsymbol{W}, \boldsymbol{\Theta}\}$ can take: (a) the range of values are grouped in two intervals $[0, a]$ and $[-a, a]$. (b) three range of values of the form $[-a, a]$ with $a \in 1, 5, 10$.

monotonically growing of the activation function together with the positive synapsis, making impossible to evolve the structure to reach its maximum. In addition, the deviation of the data is too small for all the structures and it is less than $IQR < 0.02$ (see Figure 4.15(a)). Thus, it can be assumed that for this type of ranges, the neural structure does not evolve properly and this type of configuration can be discarded.

On the other hand, for structures in which $\{\boldsymbol{W}, \boldsymbol{\Theta}\} \in [-a, a]$, the maximum fitness is almost reached. The maximum whisker for all of them is between 0.9 and 1.0 fitness value. It can be observed that the median is slightly higher for structures with only one layer rather than the ones with two layers. Moreover, the data deviation presented in one layer structures is less than half the deviation of the two hidden layer structures. Structures with two hidden layers have very similar performances and they are not so different to each other. On the other hand, the increase of neurons in one hidden layer structures, reduces the variance of the result.

Structure	Min	Q1	Median	Q3	Max	IQR	Mean	SD
1L2	81.92	87.16	90.28	91.41	92.95	4.25	88.51	4.46
1L3	82.63	88.05	90.98	91.85	93.08	3.80	89.29	4.17
1L4	85.34	89.08	90.99	91.89	93.16	2.81	89.84	3.36
2L22	71.51	83.88	88.88	92.12	94.12	8.24	86.93	6.82
2L23	70.16	83.23	89.78	92.49	94.14	9.26	87.04	7.00
2L24	74.29	84.80	90.43	92.48	94.68	7.69	87.54	6.98
2L32	72.67	84.47	89.17	92.42	93.73	7.95	87.33	6.40
2L33	72.02	84.31	89.87	92.55	94.26	8.24	87.47	6.83
2L34	76.75	85.72	90.18	92.67	94.18	6.95	88.35	5.65
2L42	72.32	83.96	89.51	92.36	94.25	8.40	87.22	6.57
2L43	75.56	84.67	90.81	92.71	94.68	8.04	88.09	6.46
2L44	79.53	87.28	91.45	92.73	94.76	5.45	89.07	5.62

Table 4.3: Statistical data of the simulation performance for all the structures where $a = 5$. The fitness value is expressed in %.

Figure 4.15(b) shows the performance for each value of $a \in 1, 5, 10$ for the range of values of the form $[-a, a]$. For all the structures, the best performance is achieved for the range of values in which $a = 5$. The worst performance is for the one in which $a = 10$, it presents lower median values than the other two intervals. The reason is the length of the interval, now there are more parameters to choose and it makes difficult the GA to find a better solution. In spite of having more dispersion than the rest of values ranges, $a = 1$ have a higher performance than $a = 10$ since the median for all the structures is higher. However, it is still lower than the performance of $a = 5$ for all the structures.

Therefore, the best performance is the one in which $\{\boldsymbol{W}, \boldsymbol{\Theta}\} \in [-5, 5]$. Statistically all of them are very similar since all of them are nearly close to the same values, but one layer structures present an IQR lower than the two layer structures. In order to help with the decision of which structure presents a better performance in this interval, the numerical information has been gathered in Table 4.3. All the structures present very similar values, however neural structures with only one hidden layer has higher minimum values than the two layers structures. In addition, the medians are close to the maximum of the boxplot, but the deviation of the data is lower in one layer structures. So, the complexity introduced by adding a second hidden layer provides slight performance improvements. In contrast, it increases the computation of the network, which in this case is oversized for the problem at hand. Thus, simpler structures are chosen for this problem and a structure with one hidden layer is selected.

Among single hidden layer structures, the performance of the one with four neurons in the hidden layer is clearly the winner of the three of them. There are not significant difference among them according to Table 4.3, but in the overall performance of the three value ranges, $1L4$ is clearly the best of the three. Finally, the best fitted structure for the problem is $\boldsymbol{1L4}$ with $\{\boldsymbol{W}, \boldsymbol{\Theta}\} \in [-5, 5]$. Thus, the selected structure is not too complex but enough to reach higher values of fitness in this interval and relative easily due to the data dispersion is narrow.

Before analysing the influence of the parameters of the GA in the performance of the selected structure, it is interesting to show the trends in evolution. Figure 4.16 shows the best, the average best, the average, the worst and the average worst fitness values of the 32400 simulations for this structure. It shows that for the $[0, a]$ interval, the structure did not evolve along the time since the average values are almost the same and the best one is very close to the general average. This behavior for $[0, a]$ confirms the performance observed from the different structures in Figure 4.15(a), in

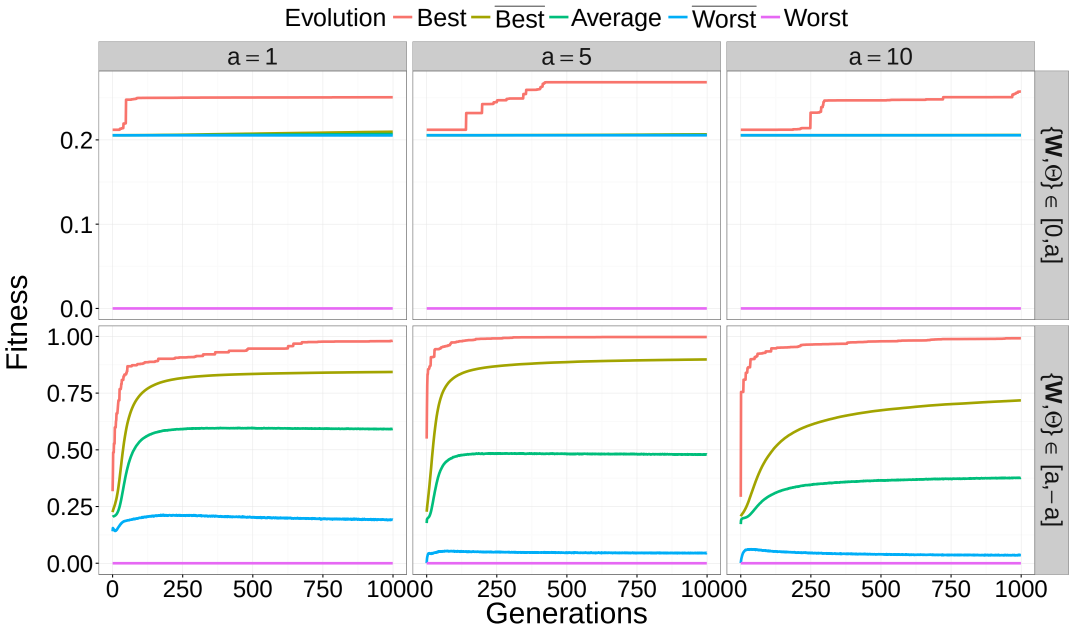


Figure 4.16: Fitness function representation to evaluate the derivative of the environment signal.

which there is almost non existing evolution. For the value ranges of the form $[-a, a]$, the best fitness almost reaches the best value in all of them. However, for $a = 1$ this evolution is slower than the rest since it reaches a plateau value almost at the end of generations and the best value is slightly below the rest of intervals. In addition, the minimum generation value of fitness for $a = 1$ is greater than the other two, but it decreases as the number of generations and the optimal solution has been found since more individuals perform near the best value achieved. For the other two values of a, the best fitness value grows logarithmically. For $a = 5$, the growth is more pronounced than $a = 10$ and rapidly reaches a plateau value close to the maximum fitness value. In average the best value for $a = 5$ behaves as the maximum best fitness, reaching also the highest values. In contrast, the best fitness value growth for $a = 10$ is more pronounced than $a = 1$, but in average this best fitness value is still growing because 1000 generations were not enough to reach a plateau value in the 5400 simulations under these conditions.

Thus, the best choice of parameters for our neural controller are in the interval $\{\boldsymbol{W}, \boldsymbol{\Theta}\} \in [-5, 5]$, at least under the conditions imposed for the training. Figure 4.16 shows that this range of values reaches a stable plateau value rapidly and gets the closer value to the maximum fitness. However, it is still unknown how the evolution for the structure and parameters selected changes with respect to the GA parameters.

Influence of GA configuration

The neural structure selected consists of a single hidden layer with 4 neurons inside it. The range of values that the free parameters can take are inside the interval $[-5, 5]$. Now, the different GA parameters have to be selected in order to understand their influence in the training of the CTRNN selected. The election of the parameters resides in the performance of the neural structure based on the different configurations that the 3 free parameters of the GA (population, mutation and fitness function) can take.

Firstly, the performance of networks is compared based on the size of the population and the mutation rate. Figure 4.17 shows the effects in the performance of the structure because of these 2 parameters. It has been represented the

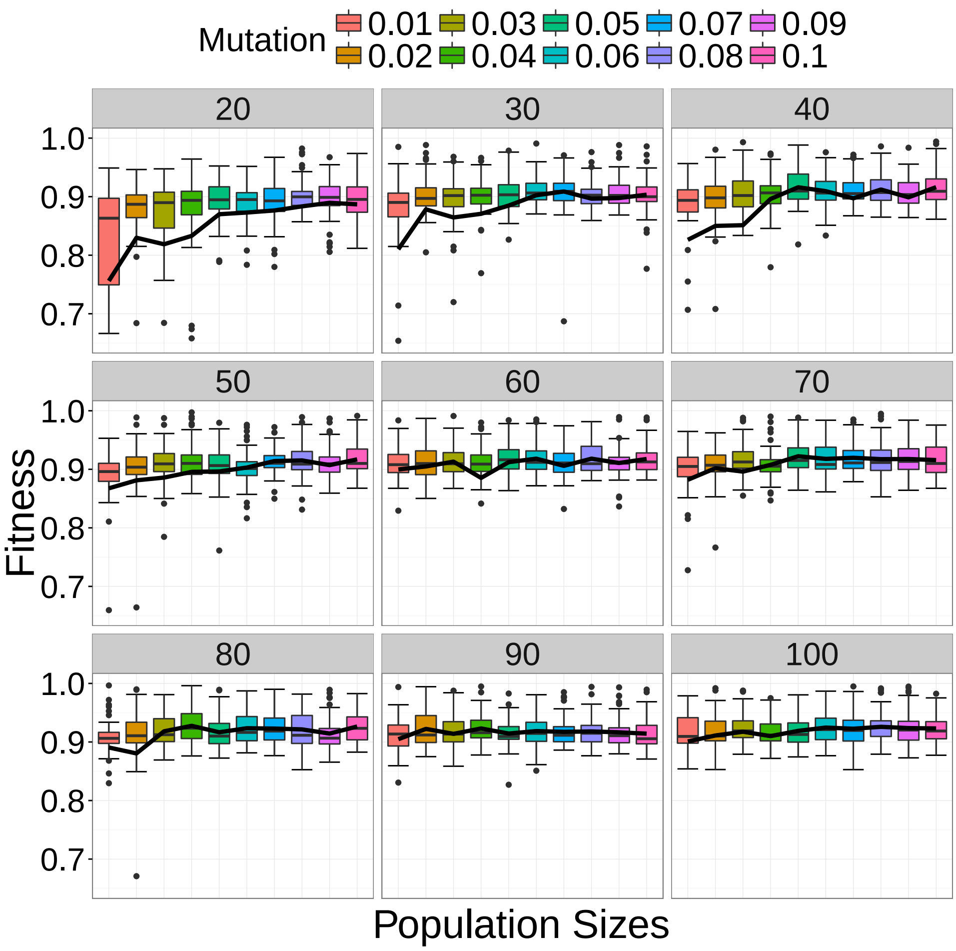


Figure 4.17: Fitness performance for the selected structure with different population sizes and mutation rates. Above each panel the size of the population has been indicated and each boxplot inside the panel represent a different mutation rate. The black line represents the mean performance of each population. Some outliers are not represented in order to zoom in the boxplot representation.

population sizes versus the mutation rates grouping together the fitness values for the 2 parameters. Each panel of Figure 4.17 represents a population size. For each of them, 10 boxplots (each per mutation rate) of 60 points are represented.

The differences among the results are very short, all of them seem to have a good performance and they are pretty similar. However, the smaller differences will give some clues on how to select the best parameters of the GA. The worst results are obtained for the lowest population size and the lowest mutation rate. In general, the data dispersion is low for all the boxplots. It can be observed a growing fitness trend as the mutation rate grows. This effect occurs due to the increase of random changes in the neural parameters, so that the GA explores faster the search space and can reach the solution in less time. However, for too high mutation rates, the algorithm may have trouble finding the solution because of the random parameter changes. That is the reason why in general for higher mutations, the data dispersion found is higher than for lower ones for the different population sizes (except to a population size of 20 individuals).

Moreover, the fitness value grows as the size of the population grows. The difference of performance for lower population sizes and higher ones is less than 10 %,

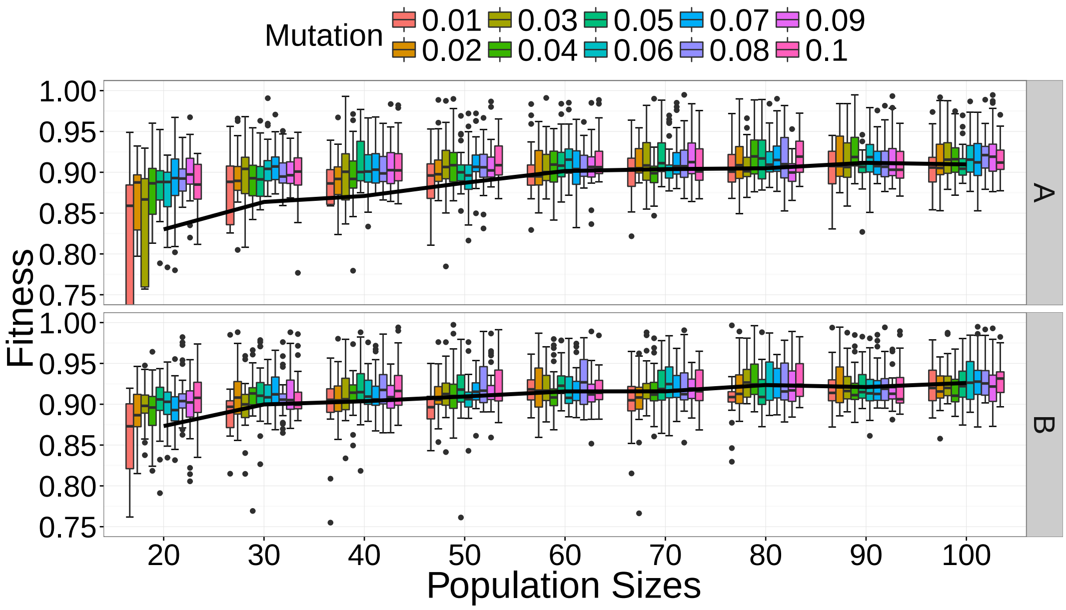


Figure 4.18: Results of the fitness performance for the three different parameters of the GA with the structure and interval selected. Panel A shows the results of the fitness function whose $\Delta s[k] = s[k+1] - s[k]$. While panel B shows the results of the fitness function for $\Delta s[k] = \left(\sum_{l=k-10}^{k+1} s[l+1] - s[l]\right)/10$. In black, it is the mean performance per block of configuration values. Some outliers are not represented in order to zoom in the boxplot representation.

so that the influence of the population size is related with the number of possibles solutions that the GA can evaluate in one generation. The probabilities of finding the optimum are increased when more individuals are part of the generation, and the diversity introduced for the genetic operations is also increased. That is the reason to use higher values of population sizes, but also larger populations will increase the operations done during a generation and the speed of the simulation will decrease.

Comparing both parameters at the same time, it can be stated that the larger the population size and the mutation rate are, the higher the fitness value is. In addition, in low populations of individuals, higher mutation rates achieve same performance levels as large populations of individuals with low mutation rates. So, both parameters are directly related and depending on the problem to solve, their values can be adapted without necessarily lose the performance level. In this case, it can be used high population sizes (N_p =100 individuals) and large mutation rates ($p_m = 0.1$) to obtain the best fitness values.

The last parameter to be selected is the way in which the fitness function is calculated in order to evolve the neural controller in the best way. Panels of Figure 4.17 are divided into two, depending on the approximation used. The results are gathered in Figure 4.18, in which panel A contains all the result for a $\Delta s[k] = s[k+1] - s[k]$, and panel B contains the result for $\Delta s[k] = \left(\sum_{l=k-10}^{k+1} s[l+1] - s[l]\right)/10$. First option computes directly the difference with the previous value of the environment signal, while option B does the average value of the environment signal in a window of 10 samples. In both cases, the fitness performance grows with the population size and the mutation rate. So, large populations of individuals with high mutation rates reach better performance values than smaller populations with lower mutation rates. Moreover, in both cases, the dispersion of the data decreases as the population size and the mutation rate increases.

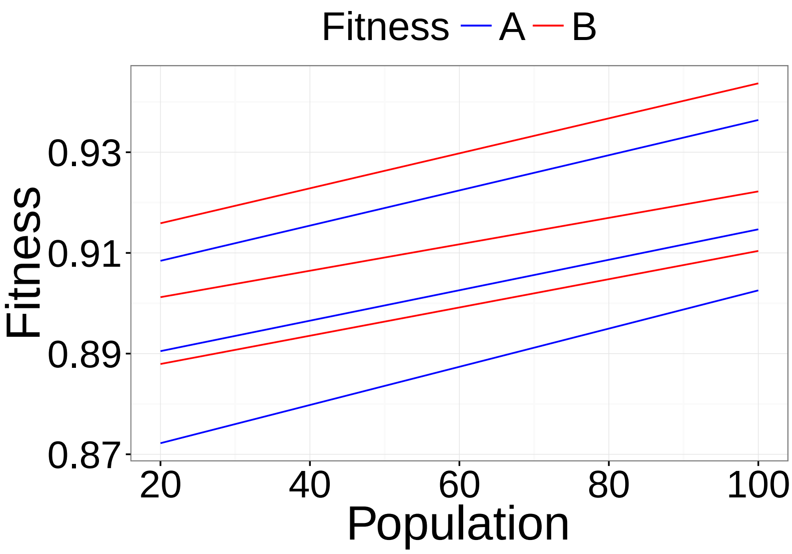


Figure 4.19: Fitness tendency of the maximum, median and minimum for the two different options, A and B, fitness evaluation. Option A contemplates that $\Delta s[k] = s[k+1] - s[k]$, whereas for option B, $\Delta s[k] = \left(\sum_{l=k-10}^{k+1} s[l+1] - s[l]\right)/10$.

However, the results for the Fitness A are lower than the results for Fitness B. The mean performance for option A begins in a value lower than 0.85 and increases its value until it surpasses 0.9, being close to it. On the other hand, option B presents a smooth ascent beginning approximately in 0.87 until it reaches the end of the simulation with a mean value that also surpasses the 0.9 performance, approximately 0.92. Option A presents a higher slope for the mean of the different simulation configurations. In spite of being closer, option B presents better results than option A. However, it is difficult to appreciate the differences for the maximum and the minimum in Figure 4.18, so that the maximum, median and minimum tendencies have been calculated for the two fitness options.

Figure 4.19 represents the different tendency of quartiles for the data from Figure 4.18. The data is sorted by the population growth. In the case of $Q3$, both fitness options are parallel but still the maximum values are achieved by option B. On the other hand, the median and $Q1$ are not parallel, so that the differences among the two options decreases as the population sizes increases. In both cases, the slope of option B is less than the slope of option A. However, option B obtains always better results than option A. Thus, option B will be selected as the fitness function to train our network structure.

Summary of selected parameters

In conclusion, all the referred neural architectures have just been trained to prove the best set of parameters for the algorithms. The selected parameters are those whose performance is the best of all the possible configurations. Hence, the architecture with the best performance and complex enough to solve the problem, is the one with a single hidden layer and 4 neurons inside it ($\boldsymbol{1L4}$). The best range of values for the free parameters of the neural structure is the one in which positive and negative values can be taken, because inhibitory and excitatory synapsis can be arranged between the different neurons. The interval selected is $\{\boldsymbol{W}, \boldsymbol{\Theta}\} \in [-5, 5]$.

Then, it has been discussed how the different configurations of the GA affects the performance of the neural structure. Therefore, after the simulation experiments, it has been found that the higher populations sizes are, the higher the fitness performance is. The reason is that more individuals are tested each generation,

CTRNN	
Structure	$1L4$
$\{\boldsymbol{W}, \boldsymbol{\Theta}\}$	$[-5, 5]$
GA	
Population size	100
Mutation rate	0.1
$\Delta s[k]$	$\left(\sum_{l=k-10}^{k+1} s[l+1] - s[l]\right)/10$

Table 4.4: Summary of the parameters selected after the simulations for both algorithms.

making easier to find the best solution from the diversity of the population. In addition, the same effect happens with the mutation rate, the higher it is, the higher the fitness performance is. And finally, from the two possible fitness options proposed, the instant derivative of the environment signal has lower performance than the cumulative approximation of 10 previous samples of the environment signal. Thus, the final configuration of the GA consists of a population of 100 individuals, a mutation rate of 0.1 and the fitness uses $\Delta s[k] = \left(\sum_{l=k-10}^{k+1} s[l+1] - s[l]\right)/10$. All the final configurations of the CTRNN and the GA are gathered in Table 4.4.

4.5 Post-evaluation

After evolution, it is necessary to check that the behavior of the evolved CTRNN with the selected parameters $\{\boldsymbol{W}, \boldsymbol{\Theta}\}$ corresponds to the objective evaluated by the fitness function. Thus, a process of post-evaluation is performed to test that the evolution was correct. This process will consist of studying the output of the evolved neural controller for the characteristics of the environment described in Figure 4.9. In order to carry out the post-evaluation, some figures of merit have to be declared. These figures will help to quantify the effect of the neural controller output in the environment in order to smooth its shape and tend to be constant.

Thus, 7 were the selected coefficients for the assessment of the signal. They measure different aspects of the signal to quantify its flatness and the time taken to converge to the solution. In order to analyze the waveform of $s[k]$, 4 central moments have been used which are computed in terms of the signal mean value. The next 2 coefficients are relationships with different parameters of the signal to evaluate its variability. And the last one is a measure of time to know when the algorithm reaches a stable operating mode. So, these coefficients are as follows:

- *Mean* ($\mu_0 = \mu$). The mean is the central value that in average the signal takes through a time period. This measure will give information about where the signal will be on average and will also serve as the reference value to develop the next central moments around it. The post-evaluation signals are greater than zero, so this value will be also greater than zero. This parameter is calculated as in Equation 4.26.

$$\mu_0 = \mu = E[X] = \frac{1}{K}\sum_{k=1}^{K} s[k] \tag{4.26}$$

- *Variance* ($\mu_2 = \sigma^2$). The variance is a measure of the dispersion of the data around a value, in this case μ. For us, it will indicate how far the points of $s[k]$ are spread out from its average. Higher values of σ^2 will indicate that the variability of the signal is too high, while a zero value indicates that all values will be the same, so that $s[k]$ will be constant. σ^2 is the second central moment and its mathematical expression is as follows:

$$\mu_2 = \sigma^2 = E\left[(X - E[X])^2\right] = \frac{1}{K}\sum_{k=1}^{K}(s[k] - \mu)^2 \tag{4.27}$$

- *Skewness (μ_3).* This parameter measures the asymmetry of the data, if it is above or below μ. Thus, it indicates how the signal is distributed around its average in a time period. It can be interpreted for positive values as the signal is above μ most of the time. Whereas if it is below μ, it indicates that most of the time the signal is below the data. In case of 0, the data is around the mean. This is the third central moment and it is calculated as:

$$\mu_3 = E\left[(X - E[X])^3\right] = \frac{\frac{1}{K}\sum_{k=1}^{K}(s[k] - \mu)^3}{\sigma^3} \tag{4.28}$$

- *Kurtosis (μ_4).* This is another measure of the form of the signal as μ_4. In this case, the kurtosis is studying the dispersion of a signal related to the average through the points closest to it compared with those points of the distant ends. Therefore, a high value of μ_4 means that there are lots of points around the mean but at the same time lots of them are also in the ends of the signal. On the contrary, a low value means that all the points are concentrated in the average of the signal. The mathematical expression is:

$$\mu_4 = E\left[(X - E[X])^4\right] = \frac{\frac{1}{K}\sum_{k=1}^{K}(s[k] - \mu)^4}{\sigma^4} \tag{4.29}$$

- *Coefficient of variation (c_v).* This parameter establishes a relationship between the average of the signal and its variance. When the data is around the mean, this coefficient is next to 0 and otherwise the data will present mean deviations. This coefficient is described mathematically in Equation 4.30.

$$c_v = \frac{\sigma}{\mu} \tag{4.30}$$

- *Crest factor (C_f).* The last parameter related with the form of the signal is the crest factor. This factor measures the waveform through the ratio of maximum values of the signals compared to its effective value. Thus, it indicates how high the peaks are present in the form of the signal. A C_f of 1 indicates that the signal does not have any present peak and it is constant, whereas higher values indicates how accused those peaks are. This factor compares the maximum of a signal with respect to Root Mean Square (rms) of the signal for a time period (see Equation 4.31).

$$C_f = \left.\frac{|s|_{peak}}{s_{rms}}\right|_K \tag{4.31}$$

- *Time of convergence (t_c).* The last parameter used is the convergence time of the neural controller to flatten the environment signal. It is a measure of the stability of the algorithm indicating when the signal becomes stable and abandons the transitory state. Hence, it is considered that the environment has reached a stable state when the σ^2 is less than an established minimum during a period of time. This is expressed in Equation 4.32.

$$t_c = |\sigma^2| < \sigma^2_{min}\Big|\, K \tag{4.32}$$

Signal	μ	σ^2	μ_3	μ_4	c_v	C_f	t_c
$z_1[k]$	1.000	0.00	0.00	0.00	0.00	1.0000	14
$z_2[k]$	0.999	$2.98 \cdot 10^{-7}$	$4.77 \cdot 10^{-7}$	$-1.13 \cdot 10^{-6}$	$5.46 \cdot 10^{-4}$	1.0008	25
$z_3[k]$	1.003	$1.59 \cdot 10^{-3}$	$-1.67 \cdot 10^{-5}$	$9.65 \cdot 10^{-6}$	$3.98 \cdot 10^{-2}$	1.0880	56
$z_4[k]$	0.998	$4.65 \cdot 10^{-5}$	$-2.38 \cdot 10^{-7}$	$2.98 \cdot 10^{-7}$	$6.83 \cdot 10^{-3}$	1.0101	28
$z_5[k]$	1.003	$3.67 \cdot 10^{-5}$	$-9.54 \cdot 10^{-7}$	$1.91 \cdot 10^{-6}$	$6.04 \cdot 10^{-3}$	1.0149	35
$z_6[k]$	0.994	$6.68 \cdot 10^{-2}$	$8.97 \cdot 10^{-4}$	$6.37 \cdot 10^{-2}$	$2.60 \cdot 10^{-1}$	1.9468	24

Table 4.5: Summary of the post-evaluation results for the 6 different signals.

The different measurements have been introduced to evaluate the response of the evolved neural controller. The best chromosome with higher fitness value has been chosen for the neural architecture of 1 hidden layer and 4 neurons inside it with its free parameters of the form $\{\boldsymbol{W}, \boldsymbol{\Theta}\} \in [-5, 5]$. To begin the post-evaluation, it is necessary to select several waveforms of $z[k]$ that the neural controller has to smooth. 6 different signals have been selected, each with different characteristics. These signals are chosen in order to evaluate the adaptability and the ability of generalization of the network. Their mathematical expressions are gathered in Equation 4.33.

$$z_1[k] = 0 \quad (4.33a) \qquad z_2[k] = \sin(f \cdot k) \quad (4.33b)$$

$$z_3[k] = \sin(2 \cdot f \cdot k) \quad (4.33c) \qquad z_4[k] = \sin(f \cdot k/2) \quad (4.33d)$$

$$z_5[k] = \begin{cases} a \cdot k & \text{if } 0 \leq k < K/2, \\ 1 - a \cdot k & \text{if } K/2 \leq k \leq K. \end{cases} \quad (4.33e) \qquad z_6[k] = \begin{cases} 0 & \text{if } 0 \leq k < K/2, \\ 1 & \text{if } K/2 \leq k \leq K. \end{cases} \quad (4.33f)$$

The first signal $z_1[k]$ of Equation 4.33a is a constant signal of zero value. This signal is used to test the response of the network in absence of any other element, only its own output. Then, $z_2[k]$ is a periodic sinusoid with the same frequency of the one used during the evolution (see Equation 4.33b). $z_2[k]$ was chosen to test that the fitness value obtained after the evolution was correct and it is able to compute the opposed derivative of the input signal. $z_3[k]$ and $z_4[k]$, whose expressions are gathered in Equations 4.33c and 4.33d, are both sinusoid waveforms, but with different frequencies. $z_3[k]$ has a frequency double of the sinusoid of $z_2[k]$ and $z_4[k]$ has half of the frequency. These two signals test how the network responds to changes in the speed of the waveforms. Finally, two other signals were defined that present some peculiarities in their form and in the derivative of both of them. Both are periodic but its derivative suffers some sudden changes. $z_5[k]$ is a triangular signal which is periodic and continuous but the sign of the derivative changes suddenly at the peak (see Equation 4.33e). And $z_6[k]$ is a square signal with a duty cycle of 50% (half of the period is at 0 and the rest is at 1), which is also periodic but not continuous (see Equation 4.33f). In both cases, these signals are used to analyze what happen when the derivative change ($z_5[k]$) or it is undefined ($z_6[k]$) to evaluate the response of the network to signals of different nature to the ones proposed for the training.

The post-evaluation was done by evaluating the behavior of $s[k]$ over a period of the waveform selected when the algorithm has reached its stable state. All the results to the signals described above are presented in Table 4.5. In addition, Figure 4.20 shows the different waveforms for each of the signals post-evaluated. In general, the algorithm is able to compute the derivative of the environment signal, to oppose to it and to flatten it. Moreover, it also converges in few steps to a stable state in which it will remain until the end of the simulation.

However, the results are not the same for all the signals due to its characteristics. In the case of the absence of input, $z_1[k]$, it can be observed in Figure 4.20(a) that $x[k]$ is constant and also $s[k]$ since there is no other signal. Thus, in the absence of $z[k]$, the network obtains a constant output. Table 4.5 shows that μ is located at a

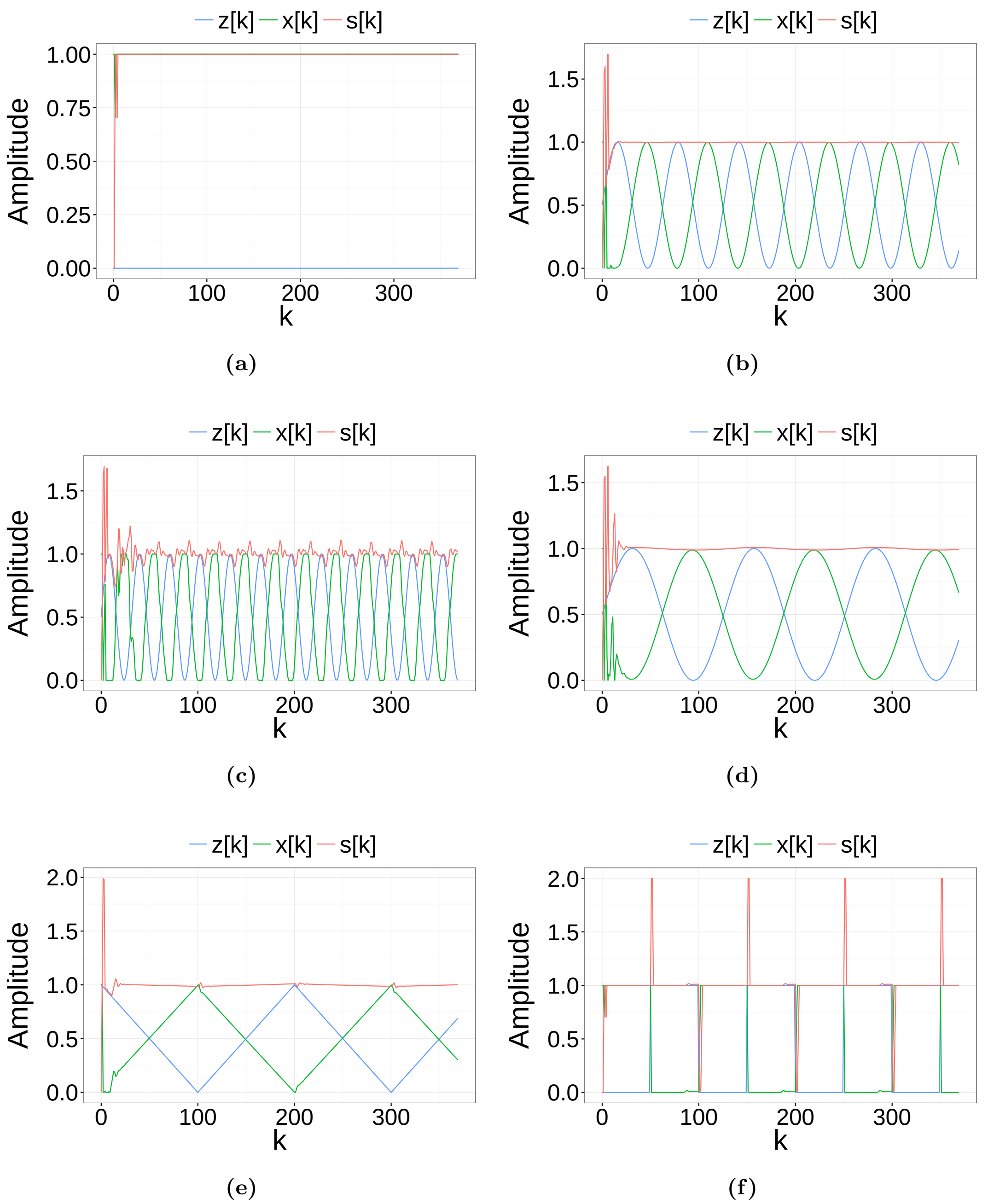


Figure 4.20: Waveforms for the post-evaluated signals selected: (a) $z_1[k]$, absence of $z[k]$, (b) $z_2[k]$, sinusoidal $z[k]$ with the same frecuency used for evolution, (c) $z_3[k]$, sinusoidal $z[k]$ with double frequency than the one in (b), (d) $z_4[k]$, sinusoidal $z[k]$ with half frecuency than the one in (b), (e) $z_5[k]$, triangular $z[k]$ and (f) $z_6[k]$, square $z[k]$. In red is the environment signal, $s[k]$, in blue is the non-controllable signal, $z[k]$, and in green is the output of the CTRNN, $x[k]$.

value of 1 and the rest of the dispersion parameters are *zero*. The C_f has the best value that it can take which is 1 because it does not present any peaks. The t_c of the

Signal	z_{C_f}	s_{C_f}	$\Delta C_f / s_{C_f}$ [%]
$z_1[k]$	–	1	0
$z_2[k]$	1.63065	1.00083	62.9302
$z_3[k]$	1.63137	1.08802	49.9397
$z_4[k]$	1.63166	1.01013	61.5298
$z_5[k]$	1.32944	1.01495	30.9859
$z_6[k]$	2.27303	1.94682	16.7563

Table 4.6: Comparison of C_f with and without the evolved neural controller.

network is quickly, in 14 time steps it reaches the stable state in which it will remain until the end.

The next signal, $z_2[k]$ is the same signal used for the training, so it is expected a high performance. Table 4.5 shows that $s[k]$ is centered in 1 again and there is not any dispersion around μ. Its σ^2 is negligible, the skew of the signal is near zero and they are very concentrated in the mean since the kurtosis is also negligible. The c_v also indicates that the dispersion of the data is small and C_f confirms the absence of peaks for $s[k]$ since is near to the optimum. t_c is again very low in 25 time steps the algorithm has already converged. The waveforms of Figure 4.20(b) show that $x[k]$ is in antiphase to $z_2[k]$ to produce a flattened $s[k]$.

$z_3[k]$ and $z_4[k]$ are again two sinusoids to prove the behavior of the network when the frequency of the sinusoid varies. In the case of $z_3[k]$, it can be observed in Figure 4.20(c) that the controller does not behave as great as in Figure 4.20(b), because it presents a ripple in $s[k]$. In Table 4.5, the moments σ^2, μ_3 and μ_4 are higher than ones for $z_2[k]$ so the data dispersion is bigger. And also the form of the waveform presents some peaks that are also present in the value of C_f. In addition, t_c is approximately double the one of $z_2[k]$. On the contrary, when the frequency is lower, the results are very similar to the one obtained for $z_2[k]$. Low values for the dispersion coefficients, σ^2, μ_3, μ_4 and c_v. And also a very close value of C_f to 1, so that it has not any pronounced peak. t_c is also as low as $z_2[k]$. Thus, the neural controller behaves similar for frequencies less than the one selected for the evolution. For frequencies higher than the one selected in the evolution they reduce the variability of the signal but they present some peaks in the resultant waveform.

$z_5[k]$ has little dispersion around the mean as the value of the σ^2 and c_v are low (see Table 4.5). In addition, the data is concentrated in the mean and with no skew above or below due to moments μ_3 and μ_4 are also low. The waveform is also really flatten as it can be seen in Figure 4.20(e) and checked with a C_f close to 1. And the convergence time is low, only 35 time steps. On the other hand, the performance of $z_6[k]$ is the worst compared to the rest of signals. There is more dispersion around the mean because the σ^2, μ_4 and c_v are higher compared to the rest. The data is not skewed because it presents a relative low value. About the form in Figure 4.20(f), the presence of peaks can be observed at the point in which the signal changes of state. Therefore, C_f has a high value far from 1 because of the presence of those peaks. However, the response of the network is very fast and in only 24 steps the network has converged.

Table 4.6 shows a comparison summary of the algorithm application. In the case of $z_1[k]$, there is no improvement due to the absence of signal at the input of the network. There is a reduction of C_f in all cases. In the best of the cases (for signals $z_2[k]$ and $z_4[k]$), a reduction of the 60 % was achieved. Whereas in the worst case (for signal $z_6[k]$), a reduction of 15 % was achieved. Thus, the application of the algorithm always reduces the peaks, increasing the flattening of the $s[k]$.

A last trial was carried out in order to test how the neural controller performs with a signal similar to the grid aggregated consumption. Thus, an artificial grid

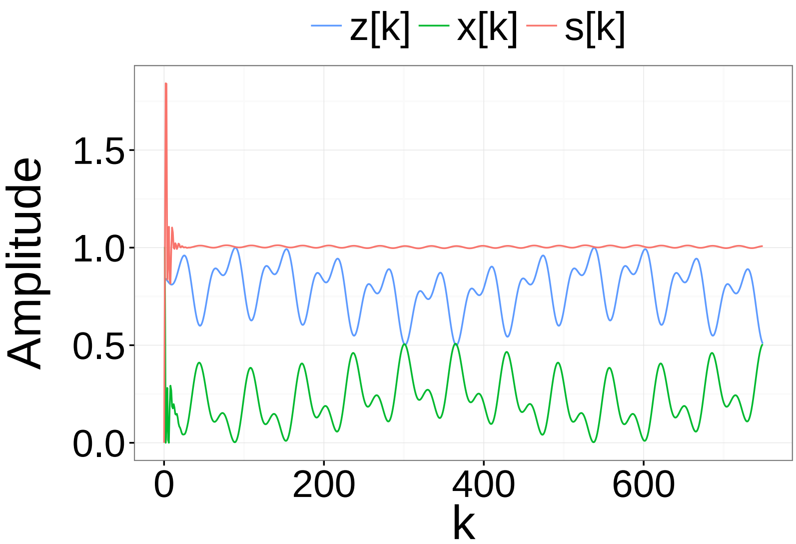


Figure 4.21: Result of the post-evaluation for a $z[k]$ with a waveform similar to the grid. In red is the environment signal, $s[k]$, in blue is the non-controllable signal, $z[k]$, and in green is the output of the CTRNN, $x[k]$.

signal is made based on the principal components of its spectrum. It is used the sum of three sinusoids with the weekly, daily and half daily frequency components. The mathematical expression is gathered in Equation 4.34.

$$z_7[k] = A_1 \cdot \cos(14 \cdot f + \varphi_1) + A_2 \cdot \cos(7 \cdot f + \varphi_2) + A_3 \cdot \cos(f + \varphi_3) + \mu \quad (4.34)$$

Hence, each sinusoid possesses a frequency in each component and the amplitude of $z_7[k]$ is between 1 and 0.5 since the consumption of a real grid is greater than 0. The application of the controller to this signal can be observed in Figure 4.21. It can be observed that the result is as expected with the previous signals, the network is able to compute the derivative, change its sign and be in antiphase to $z_7[k]$ producing an almost constant $s[k]$. μ of $s[k]$ is in 1.004 and σ^2 of the signal is $1.84 \cdot 10^{-5}$, so $s[k]$ does not present much variation and is close to the μ. $s[k]$ is concentrated in the value of μ since the skew is low, $\mu_3 = -8.34 \cdot 10^{-7}$ and $\mu_4 = 3.10 \cdot 10^{-6}$. The C_f of $z_7[k]$ is also reduced, $C_f(s[k]) = 1.0078$ and $C_f(z[k]) = 1.2703$, so the reduction of applying the neural network is a 26.05 %. This factor is also closer to the ideal value of 1. Finally, the convergence time of the algorithm is 31 steps, again the CTRNN reaches quickly a stable state.

To sum up, the application of the evolved neural network always improves the behavior of the environment signal. Reducing the variability of the $z[k]$ and obtaining an antiphase signal from the CTRNN. In the case of signals with the same nature as the one evolved, the results are better than in other cases. For signals in which the derivative does not exist, the algorithm does not perform so well, but still it could also reduce the variability of the signal, converging quickly to a steady-state.

4.6 Summary and conclusion

In this Chapter, the environment and the different elements that compose this Thesis have been presented. From the perspective of this Thesis, a grid is composed by a series of facilities, which consume electricity and could have some source of local generation. Hence, they have been divided from the controllability point of view of its consumption in controllable and non-controllable facilities. The objective of this Thesis, consists of reducing the variability of the aggregated consumption of the different users. However, this problem is too complex to solve in one step so it has been simplified by dividing the environment in only two blocks, one controllable and one non-controllable.

Then, different possibilities have been considered in order to flatten the aggregation of the different users. In this way, it is thought that cancelling the derivative of the non-controllable signal will be enough to obtain a constant aggregated signal. So, the derivative algorithm followed was explained in order to create a destructive interference that will be in antiphase to the one corresponding to the non-controllable users. This approach was possible due to the properties of the CTRNN inside the controllable user. A signal processing approach was followed to understand the problem and simplify it. Rather than using the grid aggregated consumption, a stage was used in which the non-controllable demand, $z(t)$, was sinusoidal. Thus, the CTRNN has to produce the antiphase output in order to obtain an $s(t)$ flatten.

In spite of having stated the problem, the neural architecture was not clear and it was necessary to develop a series of experiments to have the best performance network. 388800 simulations were conducted to obtain the best neural architecture consisting of 1 input layer with 2 neurons, 1 hidden layer with 4 neurons and 1 output layer with only 1 neuron corresponding to the output of the neural controller taking values for the free parameters inside the interval $[-5, 5]$. To find this architecture, 12 different structures have been evolved with 6 ranges of values for the free parameters of the network. The GA used for the evolution has different sizes of populations (20–100) and different mutation rates (0.01–0.1). To evaluate the performance, a fitness function was defined using the concept of cancelling the derivative of the environment signal to obtain a constant one. Each experiment was initialized 30 times with different seeds to randomize the beginning initial conditions of the simulation.

Once the architecture and the configuration of the GA were obtained, the behavior of the best chromosome was checked with different signals. A series of post-evaluation experiments have been done in which the first 4^{th} central moments and the c_v were analyzed to know more about the dispersion of the signal around its mean, checking if it is enough flattened. Then, the C_f was evaluated to know the form of the resultant environment signal and if it presents some peaks on it. Last parameter is the time of convergence of the algorithm or the number of steps that the algorithm takes to reach a stable state. These coefficients were tested with seven different signals which possesses different characteristics: i) absence of $z[k]$, to test the stability of the architecture, ii) sinusoid of the same frequency as the one trained, iii) sinusoid of the double of frequency, iv) sinusoid half of the frequency, v) triangular, to understand the behavior when there are points with no derivate, vi) square, to understand what happens when there are discontinuities, and vii) sum of sinusoids, imitating the grid aggregated consumption in order to see the potential of the controller developed. In all the cases good results were achieved, in the case in which the derivate is not defined, v) and vi), it was able to reduce its variability but not as well as with continuous and class C^1 signals. With abrupt changes (triangular), the controller is able to produce a flattened $s[k]$, however with discontinuities (square), is almost impossible and peaks are produced at the discontinuity point. In the best of the cases, the peaks were reduced around the 60 % and in the worst case a 16 %. In the case of the artificial signal of the grid, the C_f was near to 1 and it was reduced a 26.05 %. Thus, it can be concluded that these results are good enough to follow with the next step of the problem, the collective behavior of the controllable users described in Chapter 5.

5 Collective Controller

"The ones who are crazy enough to think that they can change the world, are the ones who do" — Steve Jobs

Once the neural controller has been evolved, a part of the problem has been solved in a reduced environment. However, it is necessary to advance on the complete problem presented in Section 4.1. To that end, this Chapter goes a step further and adds more complexity to the environment by putting more individuals of the two types in which the grid is divided. The addition of more controllable and non-controllable users threatens the neural controller developed in Chapter 4. The reason is that the controller was designed in order to smooth the demand when the controllable power is half the power of the system (the other half is coming from the non-controllable user). In this way, the neural controller was able to achieve good results and the difference between valleys and peaks was reduced.

As explained in Chapter 4, the central block of the distributed system has been designed based on the tendency of the derivative. This block is a dynamic system consisting of a Continuous Time RNN (CTRNN) that behaves in antiphase to an unknown signal. Its output, added to the non-controllable signal, cancels the variance in the resultant signal and generates a constant one. Thus, a first approach consists of using this neural controller for all the controllable users, in order to generate the antiphase signal. Figure 5.1 shows the response of an environment in which around the 65 % of the signal comes from controllable sources and the remaining 35 % is from non-controllable sources. It can be observed that each individual $x_i[k]$ puts the same exact quantity of signal and their aggregation produced more peaks in the environment signal without leaving their transitory state. However, this approach may incur in a serious problem of instability depending on the size of the environment since all the neural controllers behave the same. In addition, there are some other issues related with the nature of the controller, such as the output range (between $[0, 1]$ due to the sigmoid function) or the level of the input signal to the network since its output is bounded.

On the other hand, a real grid is a heterogeneous environment in which one of the methods used to smooth the demand curve consists of the aggregation of more loads to the system and augmenting the diversity inside it. As discussed in Section 2.1.5, the consumption of residential and service sectors present a variability that repeats periodically. The habits of users are very similar and repeat along the time but they are not the same. Therefore, the use of the same neural controller for all the controllable users provide the same response for each of them. At the time to aggregate their consumption, new peaks are found when the valley of the demand curve occurs because the controller will try to displace the consumption at those points. Thus, it is necessary to coordinate the different controllable individuals to continue having a flattened curve and not to get a worse situation than the beginning.

But how can a population of neural controllers be coordinated when there is no communication among them? The answer to this question can be easily tackled when the different parts and blocks of the system are interconnected creating a complete neural system. The reason is that synaptic connections are used to regulate the pass of information along different parts of the neural system. Thus, with enough connections among the controllers, their output could be regulated with some inhibitory or

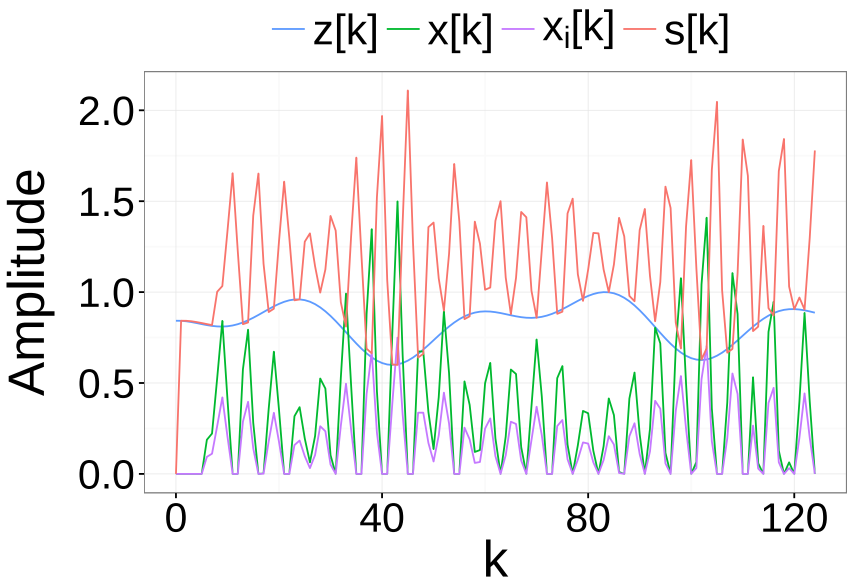
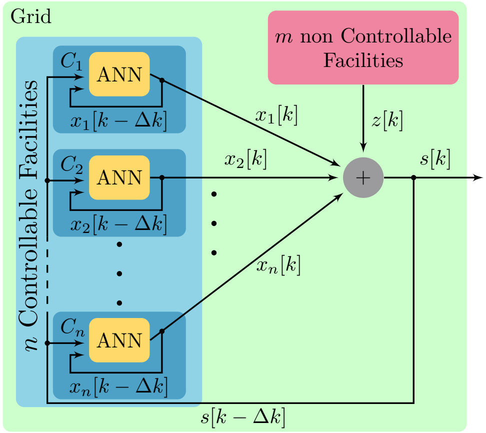


Figure 5.1: Result of applying the neural controller to each controllable individual. In blue non-controllable signal $z[k]$, in green the controllable aggregated signal $x[k]$, in purple each controllable signal $x_i[k]$ and in red the total aggregated consumption signal $s[k]$.

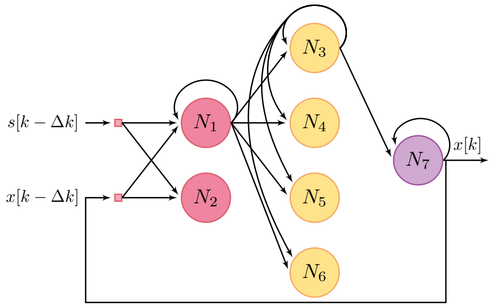
excitatory mechanisms. However, there are not any connections among them, so the problem resides in how the output of the neural controller can be modified only with the global information coming from the environment. In addition, the modification of the internal synaptic weights of the evolved neural controller will not guarantee that the network behaves the same and compute an antiphase signal. So, it is required to revise the different free parameters that are still available inside the neural controller.

Finally, it is necessary to face this new environment with a new strategy based on the evolved controller that computes the inverse derivative of the environment. Thus, a development of the evolved controller is required in this new scenario in order to adapt each controllable user and maintain the flattened curve. This development will be based on the rest of free parameters that were not used during the evolution of the CTRNN. The reason is that part of the problem is solved by using the evolved controller, but the smoothing of the signal has to be done collectively in order to fulfill the objective of this Thesis. The validity of the development is given by the number of users that can be added or the controllable sharing that can be included within the environment.

The reminder of the Chapter is as follows. In Section 5.1, the environment has been revisited to increase the level of complexity and to evaluate the possibilities of the addition of more individuals. After that, in Section 5.2, it will be introduced an algorithm for searching a solution to the coordination of the different individuals in order to achieve the flattening of the aggregated signal. Section 5.3 shows the results obtained for the different configurations of the environment in which the proposed algorithm is going to be tested. Finally, in Section 5.4, the different results obtained will be discussed during the development phase of the neural controller to reach its final stage.



(a)



(b)

Figure 5.2: Complete environment description in discrete time with the different elements that comprised it: (a) a grid environment composed by m non-controllable facilities and n controllable ones, and (b) evolved neural controller used in each controllable facility based on a CTRNN structure. The synaptic weights of each layer neuron is represented in the first neuron inside the layer.

5.1 Environment revisited

In Section 4.3, the problem was simplified in order to evolve a neural controller in a reduced part of the environment. The reason was to test the hypothesis of cancelling the derivative of the environment signal in order to obtain a constant one. Hence, the evolved neural controller was able to produce a destructive interference in antiphase, based on the tendency of the global behavior of the system. However, the system was composed by only two facilities, one for each type, being an extremely reduced scenario. In addition, a sinusoid function, with a period of a consumption day sampled in 20 minutes, was used as input during the evolution.

In the first stage of the problem, a CTRNN structure was successfully obtained in order to flatten a continuous periodic signal. However, in a real scenario, there are more than two individuals and many components can be found inside the system that consume power. Thus, the idea is to advance to the next stage of complexity that brings closer to the practical final case of study.

In this Chapter, the environment grows and consists of different individuals belonging to both types of users. As in Chapter 4, each facility only consumes power, they do not generate or store any electric power. Figure 5.2(a) shows the new environment composition, which consists of m non-controllable individuals and n controllable individuals that will oppose in antiphase to the m individuals. There is no information about each of the m non-controllable individuals, so they are grouped together, because the only available information of these users is their overall behavior. In this case, the same structure of Chapter 4 for the non-controllable users is maintained. In contrast, the environment controllable part has been divided in n facilities, since each of them will contribute to generate a response in antiphase to the m non-controllable individuals. Each of the n controllable facilities implements a neural controller, responsible for modifying the contribution of the facility to the environment. And the only information available for each of the n controllable facilities is the aggregated signal of the environment, $s[k]$. In summary, the environment is governed by Equation 5.1.

$$s[k] = z[k] + x[k] = z[k] + \sum_{i=1}^{n} x_i[k] \tag{5.1}$$

where $z[k]$ is the total contribution of the m non-controllable individuals, $x_i[k]$ is each contribution of the n controllable individuals and $s[k]$ is the environment signal which aggregates all the contributions. Both Figure 5.2 and Equation 5.1 are represented in discrete time following the same reasoning of Chapter 4.

Again, the solution to the problem is that $s[k] \to C$, where C is a constant. The controllable part of the environment is the only part accessible and adjustable from the total aggregated $s[k]$ signal. However, there is more than one individual and it is not trivial how each controllable individual should respond in order to smooth $s[k]$. Thus, based on the solution of the previous simplified environment, all controllable individuals behave as in Equation 5.2 to flatten it.

$$\sum_{i=1}^{n} x_i[k] = C - z[k] \tag{5.2}$$

However, there is no uniqueness of solutions to this problem. A priori, any combination of $x_i[k]$ that will produce a constant $s[k]$ is accepted as solution to this problem. As observed in Figure 5.2(a) and Equations 5.1 and 5.2, this environment consists of a generalization of the previous one presented in Section 4.1. Thus, it was conceived the idea of using the same neural controller designed and evolved for the previous problem in first place (see Figure 5.2(b)). In this case, all controllable users compute the opposed derivative of the input signal and respond to the environment by opposing to it and cancelling the fluctuations on it. However, if the same exact controller is used for all the n individuals, they will try to put the same contribution to the environment. This behavior will destabilize the environment signal, augmenting its variability.

The proposed problem can be tackled from a signal decomposition point of view, in which the overall behavior has to be divided in n different parts. More concretely, this problem is similar to a Blind Source Separation (BSS) problem, in which the separation of a source from the combined signal of a group is done without or with very little information about the sources or the mixing process (Haykin, 2000). However, the solutions to this problem are undetermined and the different methods that implements this technique search for a set of possible solutions in which the desired solution is included, conditioned by the problem itself. Based on these ideas, it is necessary to build an ensemble of neural networks with no communication among them that are able to self-organize and coordinate to compute the antiphase signal of the non-controllable facilities. In addition, using the idea of Chapter 4, an algorithm has to be elaborated in order to compute the opposed derivative of the environment signal collectively. For this purpose, a development of the evolved neural controller is required in order to move a step closer to the solution of the problem presented in this Thesis. This new neural controller will be referred as Evo-Devo Neural Controller (EDeNC), where evo refers to the evolutionary part of the controller and the devo to its development part. The reason is that the evolved neural controller cannot be used directly as shown in Figure 5.1 and discussed above, so that a learning process will be computed.

Following the BSS approach to the problem, numerous signals accomplish the signal decomposition proposed in first place. For example, in case the environment is composed by 2 controllable individuals (facilities, $x_0[k]$ and $x_1[k]$) and 1 non-controllable ($z[k]$), there exist multiple waveforms that can accomplish the task. Figure 5.3 shows 4 different examples in which the demand is smoothed with different behaviors of the controllable users. In Figure 5.3, it is represented the behavior of $x_0[k]$ and $x_1[k]$, the controllable aggregated consumption $x[k]$, the non-controllable demand $z[k]$ and the aggregated consumption $s[k]$. As shown in Figure 5.3(a), the first approach to divide a signal in parts will consist of equally dividing it into as many parts as necessary. However, it is unlike that two users behave exactly the same due to the characteristics of the problem. Another possible solution is that only one individual consumes everything and the other one nothing. Thus, the neural

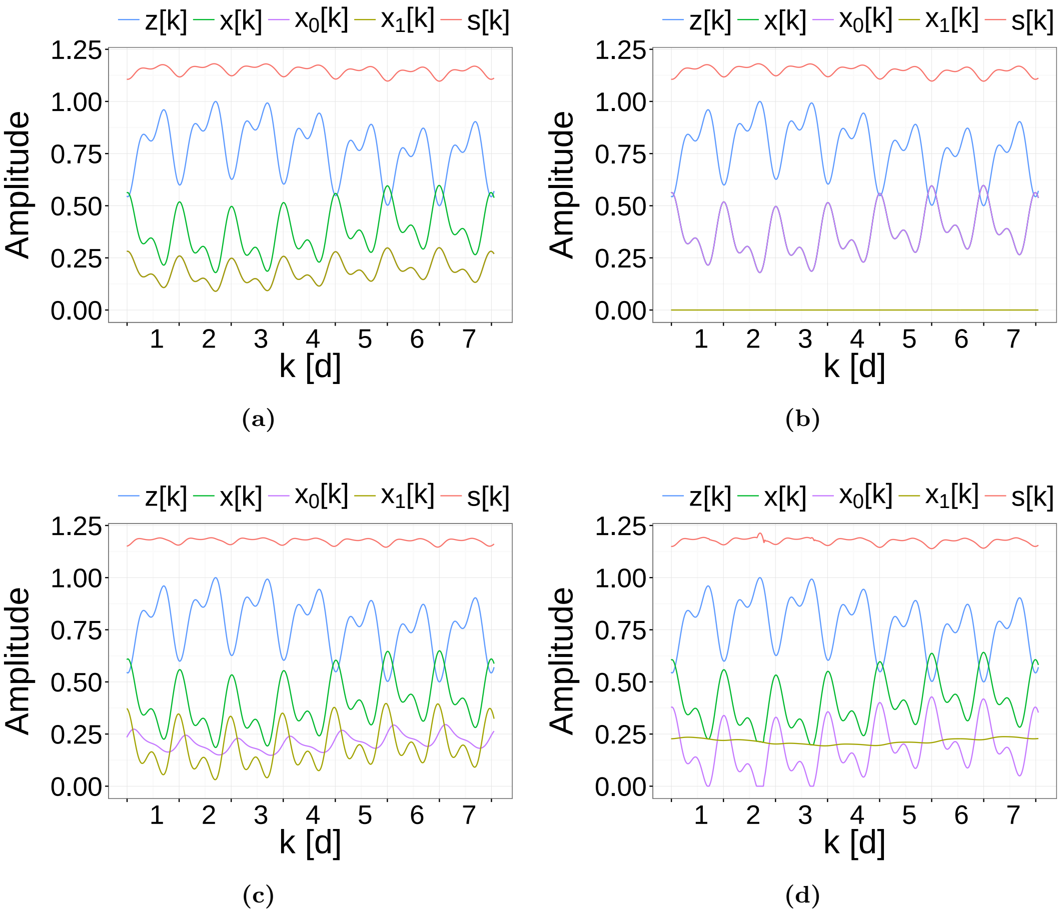


Figure 5.3: Environment composed by 2 controllable individuals, $x_0[k]$ and $x_1[k]$, with different responses that flatten $s[k]$: (a) $x_0[k] = x_1[k] = 1/2 \cdot x[k]$, (b) $x_0[k] = x[k]$ and $x_1[k] = 0$, and (c) $x_0[k]$ present a periodic waveform variable in time $x_1[k] \approx x[k]$, (d) $x_0[k] \approx x[k]$ and $x_1[k] \approx 0.25$. In blue non-controllable signal $z[k]$, in green the controllable aggregated signal $x[k]$, in purple $x_0[k]$ controllable individual, in olive $x_1[k]$ controllable individual, and in red the total aggregated consumption $s[k]$.

controller of the first facility computes directly the derivative of the environment signal (see Figure 5.3(b)). But this approach is not possible because all facilities will consume and their consumption cannot be zero all the time. On the other hand, in Figures 5.3(c) and 5.3(d), two different waveforms are observed for the two individuals whose approach differs from the previous ones. In these examples, both facilities present a consumption different from 0 and compute collectively the desirable behavior.

Different solutions can be used to solve the problem, although not all of them are feasible for this environment. Some constraints have to be considered in the design of the solution for the collective control of a group of neural ensembles. The collective behavior should take into account the following considerations taken from the real implementation of the problem:

- *Non-zero consumption.* The different facilities, which compose an electrical grid, consume power and present a demand profile that differs from 0. So it is necessary to assure that the response of the EDeNCs is different from zero for at least a period of time. Mathematically, this means that

$$\forall x_i \; with \; i \in \{1, 2, \ldots, n\}; \; \exists k \in \{1, 2, \ldots, K\} \mid x_i[k] \neq 0$$

Furthermore, this behaviour is repeated periodically by the facilities based on the consumer habits. Thus, there are various time periods (several ks) in which the consumption is non-zero. Then, solutions, such as the one of Figure 5.3(b), are not taken into account because they are not feasible for this kind of environment.

- *Heterogeneous demand profiles.* The demand profiles of the different facilities of an electrical grid are not the same. They are grouped together depending on the typical loads that generate those consumption profiles. Although they have similar loads, the demand profile is not exactly the same because users consume based on their necessities at different time periods. So, the EDeNC should produce different outputs for each controllable facility. They can be similar but not the same. Thus, solutions, such as the one of Figure 5.3(a), are possible but not reasonable for a real implementation.

- *Scalability.* Another aspect of the environment is the number of individuals who are part of it. It is necessary that the algorithm can operate independently of the number of users and adapt the output of the controllable ones to the non-controllable in order to smooth the aggregated demand curve. The proposed distributed control solution has to be able to adapt to the different possible scenarios in which the contribution of the users can vary. The environment can be composed of different combinations of m and n users, in which m can be less, equal or greater than n. Thus, the n controllable users should always react and do their best effort in order to flatten the aggregated demand curve. In addition, the algorithm should be able to react to any size of the controllable users or to the incorporation of new users into the environment.

- *Real time operation.* The last aspect to consider is its operation. It is required that the EDeNCs can adapt to the changes in the environment. So, the EDeNC learning algorithm has to adapt to those changes since the grid evolves constantly. This adaptation process should be fast enough to respond to changes in demand and able to react to any modification occurring in the environment. Then, the algorithm should operate in real time to confront these problems and the ones derived from the environment operation.

The algorithm, proposed in this Chapter, should be built taking into account these four aspects for the development of the evolved neural controllers. However, once the neural controller is evolved, the different parameters of the network are fixed and it behaves as the antiphase signal for the non-controllable users. Thus, a first approximation could be based on retuning $\{\boldsymbol{W}, \boldsymbol{\Theta}\}$ in order to adapt the output of each controller. There exist various problems to this approximation. The first one is that the Genetic Algorithm (GA) proposed in Chapter 4 cannot be used, since the operation has to be done in real time in order to vary those parameters. Secondly, each controller has to estimate a variation of its previous parameters to achieve the collective flattening of the signal. Hence, it implies that the algorithm has to be fast enough in order not to interfere with the controller output. Finally, the last problem is related with the size of the environment. Depending on the algorithm construction, adding new users could destabilize the operation due to new elements or the reactions of the algorithm could be slow.

This strategy of retuning the connections of the network is not the best strategy to tackle the problem. However, it would be more interesting to leave the network in its evolved form and change the controller output in order to avoid overlapping with the other users' output. For example, the different controller outputs can be displaced in time. Thus, they do not overlap and the controllers parameters do not have to be retuned. τ_i is the neural parameter related with time and represents the membrane time constant, related with the action potential of the neuron. The value

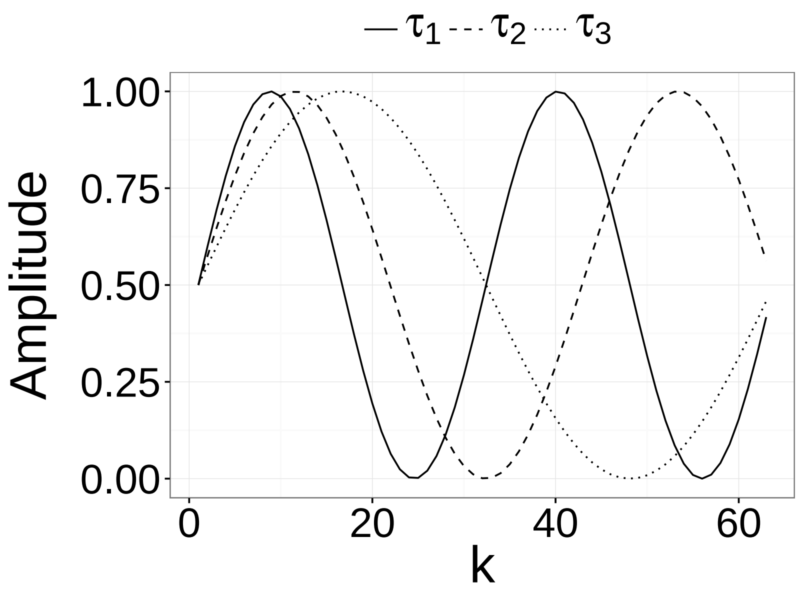


Figure 5.4: Effect of modifying the τ_i of a CTRNN whose output is sinusoidal. The three time constants are different and their relationship is $\tau_1 < \tau_2 < \tau_3$.

of this parameter represents the reaction speed of a neuron to a stimulus. Therefore, the neuron firing rate can be increased or decreased by adjusting this parameter.

In addition, τ_i was not used during the evolution. Thus, in principle, the behavior of the network will not be affected by changing τ_i. It will only slow down the network output and rescale it. This new approach will be able to tackle the scalability and the processing speed problems found with the first approximation idea. Moreover, a heterogeneous response is achieved by using this approach in an easier way since less parameters are involved in the process.

5.2 τ-Learning Algorithm: coordination of neural ensembles

In this Section, the algorithm used to coordinate an ensemble of controllable users is introduced. Each user implements the same evolved neural controller developed in Chapter 4. However, the controller response is the same for all of them so it was necessary to implement an algorithm to develop the response of the neural controllers. This algorithm is based on modifying τ_i, one of the last free parameters available inside the network.

τ_i is the time constant of the CTRNN. The modifications of this parameter affect the output as it is delayed. Figure 5.4 shows an example of the modification of τ_i, where $\tau_1 < \tau_2 < \tau_3$. The same waveform was obtained for the three outputs but with a modification of the frequency. The effect of increasing the value of τ_i corresponds to slow the output of the network increasing the period of the sinusoidal. In other words, the bigger τ_i is, the slower the response of the CTRNN is and vice versa.

According to Beer (1995), changes in the time constants do not alter the number of equilibria points. However, it could change their behavior and affect the stability and type of equilibria. Thus, it is necessary to take this into account to know which values τ_i can take. In Section 5.2.1, an analysis is done to study the possible values of τ_i that make the system stable.

Therefore, the output of the network can be displaced by changes in the value of τ_i. Moreover, based on the studies of Draye et al. (1995), the adaptation of τ_i

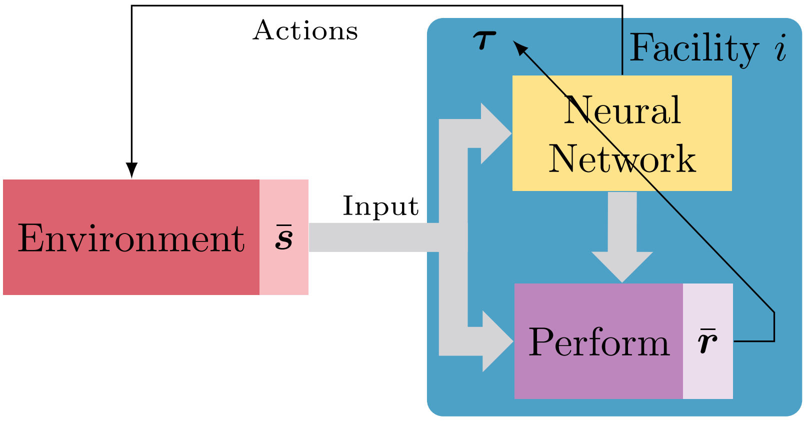


Figure 5.5: Reinforcement learning scenario to coordinate the τ_i of the different EDeNC.

value can help improving the capacities of Recurrent Neural Networks (RNNs) in the prediction of chaotic systems. By changing the values of τ_i, it can help improving the performance of the system and obtain the objective of this Thesis. But, how can an algorithm be implemented to coordinate the different τ_i? The algorithm must be able to modify in real time the value of the different τ_i from the distributed controllers around the environment. Thus, this problem can be directed as a reinforcement learning one. The reasons are that $s[k]$ contains all the information needed to evaluate the performance of the global behavior, modifying τ_i and actuating again over the environment. In Chapter 3, the typical case of a reinforcement learning scenario was defined in Figure 3.7(c). The idea consists of taking all the available information, input and reinforcement signals, from the environment in order to elaborate a reward signal and modify the output of the Artificial Neural Network (ANN). And then, the ANN acts on the environment again, repeating the cycle.

Figure 5.5 shows the adaptation of the reinforcement learning scene to our problem. Comparing Figures 3.7(c) and 5.5, both scenes are very similar. In this case, the sole information coming from the environment is $s[k]$. Thus, there is no reinforcement signal, but this information can be extracted from the input signal. There is also no observer to give the reward to the neural controller. However, inside each controllable facility, there is a performance block. The performance block elaborates the reward signal that will alter the status of the neural controller. Specifically, this block is in charge of modifying the value of τ_i depending on the $s[k]$ coming from the environment and the actions of the neural controller. Finally, once the reward or the punishment is applied to the neural controller, it will act in the environment and the cycle begins again.

After introducing the learning paradigm followed, it is necessary to develop the performance block in order to analyze the displaced output of the different neural controllers. The final objective is to achieve an environment signal that meets $s[k] \to C$. A first idea could consist of reducing the variability of the signal ($\sigma_s^2 \to 0$) in order to make that $s[k]$ tend to its mean ($s[k] \to \mu_s$). Thus, μ_s and σ_s^2 can be obtained from the input signal in the performance block. However, as a real time algorithm is required, it cannot extract these two parameters from the complete history of the signal each time step by two reasons. The first one is the computation required each time step as the number of samples increase each time step. And the second one is that distant events in the past have not excessive relevance when taking the actual decision. Thus, it is necessary to evaluate them in a window of time, W_K, and take the actions necessary to modify the behavior of the controller.

An example of these parameters is shown in Figure 5.6. In this Figure, it can be observed that $s[k]$ (in black) presents different peaks and consequently a $\sigma_s^2 \neq 0$. The μ_s for the portion of signal inside the window is shown in blue. And in green,

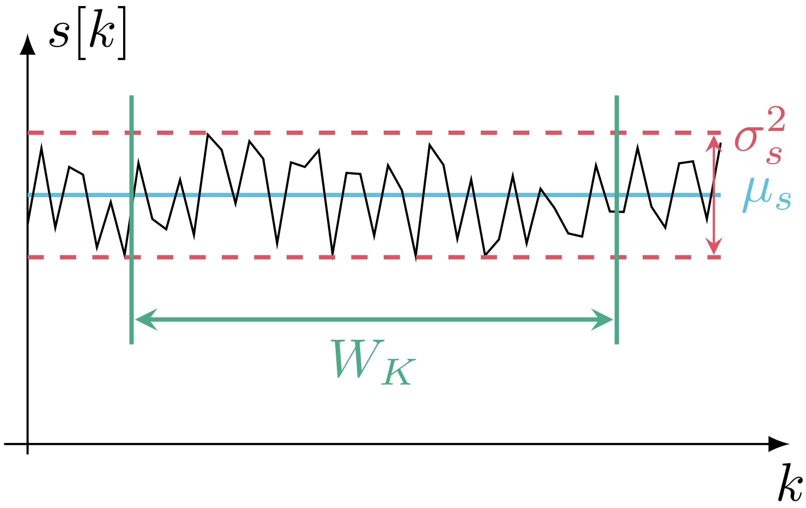


Figure 5.6: Environment signal through time, represented with its mean and variance evaluated in a window of time.

the time window is represented in which the different parameters of the signal will be determined.

A first idea to build this learning algorithm is based on reducing the peaks that $s[k]$ presents, getting closer to μ_s. So, Equation 5.3 defines an error measure based on the differences between them.

$$E_s = \sum_{k \in W_K} \mu_s - s[k] \tag{5.3}$$

where, μ_s is the mean of $s[k]$ but evaluated only inside the time window W_K, i.e.

$$\mu_s = \frac{1}{W_K} \sum_{l \in W_K} s[l] \tag{5.4}$$

Thus, the local performance of the neural controller is evaluated through the error E_s of Equation 5.3. This measure is used by the algorithm as an indicative of how well the controller is cancelling the non-controllable demand. The algorithm tries to reduce E_s in order to decrease the variability of the signal. Mathematically, this implies that the algorithm is looking for,

$$E_s \to 0 \Longrightarrow \sigma_s^2 \to 0 \tag{5.5}$$

However, it can be only tackled from a local perspective. Hence, the algorithm will minimize the E_s with respect to the local τ_i of the neural controller since it is the only available parameter. Then, the algorithm uses a gradient descent technique inspired by the supervised learning algorithms in order to minimize the error. It will compute the partial derivative of the error with respect to the partial derivative of the τ_i. The minimization of this behavior is expressed in Equation 5.6

$$\min_{\tau_i} E_s = \frac{\partial E_s}{\partial \tau_i} = \frac{\partial}{\partial \tau_i} \left(\sum_{k \in W_K} \mu_s - s[k] \right) = \sum_{k \in W_K} \frac{\partial \mu_s}{\partial \tau_i} - \frac{\partial s[k]}{\partial \tau_i} \tag{5.6}$$

Thus, the minimization of E_s has two parts, the first one corresponds to the μ_s of the signal and the second one corresponds to each point of the signal. Therefore, the problem is going to be solved by each part at a time. Firstly, the part related to μ_s is analyzed. Then, if Equation 5.4 is substituted in Equation 5.6, it is obtained

$$\frac{\partial \mu_s}{\partial \tau_i} = \frac{\partial}{\partial \tau_i}\left(\frac{1}{W_K}\sum_{l\in W_K} s[l]\right) = \frac{1}{W_K}\sum_{l\in W_K}\frac{\partial s[l]}{\partial \tau_i} \tag{5.7}$$

The result of Equation 5.7 is equal to the second part of Equation 5.6, so for both of them the partial derivative of $s[k]$ regarding τ_i is computed. In order to solve this partial derivative, Equation 5.1 will be substituted in the second part of Equation 5.6 and in Equation 5.7. The result is shown in Equation 5.8.

$$\frac{\partial s[k]}{\partial \tau_i} = \cancelto{0}{\frac{\partial z[k]}{\partial \tau_i}} + \frac{\partial}{\partial \tau_i}\left(\sum_{j=1}^{n} x_j[k]\right) = \sum_{j=1}^{n}\frac{\partial x_j[k]}{\partial \tau_i} = \begin{cases} 0 & if \;\; i \neq j \\ \dfrac{\partial x_i[k]}{\partial \tau_i} & if \;\; i = j \end{cases} \tag{5.8}$$

As can be observed, the partial derivative of the environment signal with respect to τ_i only depends on the output of the ith neural controller which corresponds to the $7th$ neuron in the output layer (see Figure 5.2(b)). Moreover, the minimization of the error only depends on the controller whose τ_i is being analyzed since the derivative is 0 for $i \neq j$. Hence, the error is calculated with respect to the output of the controller or its $7th$ neuron output. Then, the partial derivative of Equation 5.8 is as follows,

$$\frac{\partial x_i[k]}{\partial \tau_i} = \frac{\partial \sigma_i(\nu_{i,7}[k])}{\partial \tau_i} = \sigma_i'(\nu_{i,7}[k])\cdot\frac{\partial \nu_{i,7}[k]}{\partial \tau_i} = \sigma_i'(\nu_{i,7}[k])\cdot\left(\frac{\partial y_{i,7}[k]}{\partial \tau_i} + \cancelto{0}{\frac{\partial \theta_7}{\partial \tau_i}}\right) \tag{5.9}$$

where, $\sigma_i(\nu_{i,7}[k])$ is the output of $7th$ neuron for the ith controller, $\nu_{i,7}[k]$ is the propagation rule of the ith controller for the $7th$ neuron, $\sigma_i'(\nu_{i,7}[k])$ is the derivative of the activation function, which in this case is a sigmoid function and its derivative is equal to

$$\frac{d\sigma(u)}{du} = \frac{d}{du}\cdot\frac{1}{1+e^{-u}} = \frac{e^{-u}}{(1+e^{-u})^2} = \sigma(u)\cdot(1-\sigma(u)) \tag{5.10}$$

and $y_{i,7}[k]$ is the state of the $7th$ neuron of the ith controller and the only one that has a relationship with τ_i. The neural controller is governed by Equation 4.18, so the state derivative of the last neuron of the network is as follows,

$$\begin{aligned} \frac{\partial y_{i,7}[k]}{\partial \tau_i} &= \frac{\partial}{\partial \tau_i}\cdot\left[\cancelto{0}{y_{i,7}[k-1]} + \frac{1}{\tau_i}\cdot\Delta y_{i,7}[k-1]\right] = \\ &= -\frac{1}{\tau_i^2}\cdot\Delta y_{i,7}[k-1] + \frac{1}{\tau_i}\cdot\frac{\partial \Delta y_{i,7}[k-1]}{\partial \tau_i} \end{aligned} \tag{5.11}$$

where, the dependency with τ_i is limited to only one step despite being a recurrent equation. That is the reason why the term $y_{i,7}[k-1]$ is eliminated at the beginning of the derivative in Equation 5.11. Then, the term $\Delta y_{i,j}[k]$ corresponds to the discrete difference of the state for the jth neuron of the ith neural controller, and it is equal to

$$\begin{aligned} \Delta y_{i,j}[k] = -y_{i,j}[k] + \sum_{l=1}^{N_{pre}} w_{i,jl}\cdot\sigma_{i,l}\left(y_{i,l}[k]+\theta_{i,l}\right) + \\ + \sum_{f=1}^{N_{feed}} w_{i,jf}\cdot\sigma_{i,f}\left(y_{i,f}[k]+\theta_{i,f}\right) + \sum_{m=1}^{n_{in}} g_{i,m}\cdot I_{i,m}[k] \end{aligned} \tag{5.12}$$

where, N_{pre} is the number of output neurons of the previous layer, N_{feed} is the number of neural feedback loops with other neurons and n_{in} is the number of inputs. The output of the different neurons has been separated in order to obtain the derivative of each contribution. Finally, substituting Equation 5.12 in Equation 5.11, it is obtained

$$\begin{aligned}\frac{\partial y_{i,7}[k]}{\partial \tau_i} &= -\frac{1}{\tau_i^2} \cdot \Delta y_{i,7}[k-1] + \frac{1}{\tau_i} \cdot \frac{\partial \Delta y_{i,7}[k-1]}{\partial \tau_i} = \\ &= -\frac{1}{\tau_i^2} \cdot \Bigg[\Delta y_{i,7}[k-1] + \frac{1}{\tau_i} \cdot \sum_{l=3}^{6} w_{i,7l} \cdot \sigma'_{i,l}\left(y_{i,l}[k-1] + \theta_{i,l}\right) \cdot \\ &\cdot \left(\Delta y_{i,l}[k-1] + \frac{1}{\tau_i} \cdot \sum_{p=1}^{2} w_{i,7p} \cdot \sigma'_{i,p}\left(y_{i,p}[k-1] + \theta_{i,p}\right) \cdot \Delta y_{i,p}[k-1] \right) \Bigg]\end{aligned} \tag{5.13}$$

To sum up, the derivative of the error with respect to τ_i, after all the intermediate steps to obtain its expression, has the form presented in Equation 5.14.

$$\frac{\partial E_s}{\partial \tau_i} = \sum_{k \in W_K} \left[\left(\frac{1}{W_K} \sum_{l \in W_K} \sigma'_{i,7}(y_{i,7}[l] + \theta_{i,7}) \cdot \frac{\partial y_{i,7}[l]}{\partial \tau_i} \right) - \sigma'_{i,7}(y_{i,7}[k] + \theta_{i,7}) \cdot \frac{\partial y_{i,7}[k]}{\partial \tau_i} \right] \tag{5.14}$$

where, ${}^{\partial y_{i,7}[k]}/_{\partial \tau_i}$ is the expression calculated in Equation 5.13.

The gradient of the error has been computed, but the value of τ_i has not been updated yet. The next step of the algorithm will compute the new value of τ_i based on the gradient of the error. A first approximation to update the value of τ_i is as follows,

$$\tau_i[k+1] = \tau_i[k] + \left.\frac{\partial E_s}{\partial \tau_i}\right|_k \tag{5.15}$$

As can be observed in Equation 5.15, there is a problem with this update and it is based on the nature of the neural controllers. All the τ_i are going to have the same value each time step and still there is no coordination between the different ensembles of neural controllers. Then, a multiplier for the gradient of error can be used in order to set the learning pace of τ_i. This technique is used in different supervised learning algorithms to establish the growth pace of the parameters to adjust as shown in Section 3.5.1.1. Equation 5.16 shows this new τ_i update rule.

$$\tau_i[k+1] = \tau_i[k] + \alpha_i \cdot \left.\frac{\partial E_s}{\partial \tau_i}\right|_k \tag{5.16}$$

where, α_i is the learning rate of the gradient error for the *ith* neural controller. Normally, the learning rate in any supervised learning algorithm is a fixed number. However, in this case, it is necessary to use a random learning coefficient.

The objective of this algorithm is to reduce the variability of the aggregated signal when there is more than one controllable user. Thus, it is required that the τ_i varies quickly when the variations around μ_s are very high and slower when those variations are closer to zero. In addition, another issue to take into account is that the closer that τ_i is to the evolved value ($\tau_i = 1$), the more similar the controller output would be to the original antiphase signal. However, the only available coefficient to implement both design rules is α_i. It has to be related with σ_s^2 and try to spread the value of τ_i near to its evolved value. Then, the values of α_i are randomized by following an exponential distribution, such as the one of Equation 5.17.

Algorithm 1 High-level description of the τ-Learning Algorithm.

/* Initialize the environment */
$\sum_{i \in n} x_i \leftarrow$ CTRNN ▷ n individuals with the same controller
$z \leftarrow$ Power profile ▷ m non controllable individuals
for $k \leftarrow 1, \ldots, K$ **do**
    for all $i \in n$ **do**
        $x_i[k] \leftarrow f(s[k-1], \boldsymbol{y})$
    end for
    $s[k] \leftarrow z[k] + \sum_{i \in n} x_i$
    if $k \% W_K == 0$ **then**
        Calculate $\{\mu_s, \sigma_s^2\}$
        for all $i \in n$ **do**
            Compute ${}^{\partial E_s}/_{\partial \tau_i}$ based on Equation 5.14
            Obtain α_i following Equation 5.17
            Update τ_i based on Equation 5.16
            Assure stability of the system
        end for
    end if
end for

$$\alpha_i(x, \sigma_s^2) = \begin{cases} \sigma_s^2 \cdot e^{-\sigma_s^2 \cdot x} & x \geq 0, \\ 0 & x < 0. \end{cases}$$
$$where \;\; \sigma_s^2 = \left(\frac{1}{W_K} \sum_{l \in W_K} s[l] \right)^2 - \mu_s^2 \tag{5.17}$$

where, $\alpha_i(x, \sigma_s^2)$ is the exponential probability function for the α_i and whose mean is located at σ_s^2, which is the variance of $s[k]$ for W_K. This distribution has been selected because the higher probabilities are around the mean of the exponential distribution (in this case is σ_s^2). So most of the τ_i are going to have a similar rate of change. Moreover, being the exponential distribution centered on σ_s^2, α_i will change with its value. Fulfilling that α_i will be lower as the σ_s^2 is being reduced, the values of τ_i will change slowly, and vice versa.

To sum up, all the steps of the τ-Learning Algorithm (τLA) are gathered in Algorithm 1. First of all, the environment is initialized with m non-controllable users and the n controllable users. After that, the time starts and it will first calculate the output of each neural controller. Then, their responses are added to the environment together with the non-controllable users. When the time is of the same size as W_K, then the environment status is evaluated. Therefore, it computes the different measures to correct the behavior of the neural controllers and finally, it updates the value of τ_i. Algorithm 1 has been tested in simulations. However, before testing the algorithm, it must be analyzed the stability of the system and the values that τ_i can take.

5.2.1 Stability

Depending on the value of τ_i, there will be different stability behaviors. Thus, it is interesting to analyze the values that τ_i can take in order to maintain the stability of the neural controllers. Each neural controller is governed by a system of differential equations, in which each neuron is represented by Equation 4.15. However, Equation 4.15 is simplified in order to analyze the effect of τ_i in the stability of the ith neuron. This simplification is as follows,

$$\dot{y}_i(t) = a \cdot y_i(t) + u_i(t) \tag{5.18}$$

where, $a = -\,^1/_{\tau_i}$ and $u_i(t)$ is the rest of the terms of Equation 4.15, grouping together all the inputs to the neuron. In order to work with the neural controller, it is used in discrete time. The discretization of Equation 5.18 is particularized for only one sample ($\Delta k = 1$) as it was done in Section 4.3.2.

$$\frac{y_i[k] - y_i[k - \Delta k]}{\Delta k} = a \cdot y_i[k] + u_i[k] \underset{\Delta k=1}{\longrightarrow} y_i[k] - y_i[k-1] = a \cdot y_i[k] + u_i[k] \tag{5.19}$$

In order to analyze Equation 5.19, the Z-transform is used to pass from the discrete time to the complex frequency domain. The Z-transform is,

$$\begin{gathered}(1-a) \cdot y_i[k] = y_i[k-1] + u_i[k] \xrightarrow{\mathscr{Z}} (1-a)\, Y_i(z) = z^{-1} \cdot Y_i(z) + U_i(z) \\ H_i(z) = \frac{Y_i(z)}{U_i(z)} = \frac{1}{(1-a) - z^{-1}}\end{gathered} \tag{5.20}$$

Finally, the stability of the neuron is given by the poles of the $H_i(z)$, which are in the denominator $\mathscr{D}(H_i(z))$. Thus, the system will be stable if the poles are inside the unit circle of the complex plane. In this case, the Region of Convergence (ROC) is of the form,

$$|z| > \left| \frac{1}{1-a} \right| \tag{5.21}$$

Hence, the denominator of the ROC condition must be greater than 0, in order to meet and to assure the stability of the system. Then, substituting a by τ_i, the possible values of τ_i are obtained in order not to disrupt the stability, as shown in Equation 5.22.

$$1 - a \geq 0 \xrightarrow{a=-\,^1/_{\tau_i}} |\tau_i| \geq 1 \tag{5.22}$$

The stability of the neuron depends on the value that τ_i can take. It can be considered that τ_i belongs to the positively defined axis since the sign is given by the state of the neuron. Thus, in order to assure that the system is stable, $\tau_i \in [1, +\infty)$. And the τLA should assign values to τ_i respecting this condition.

5.3 Simulation Results

The environment is configured as in Figure 5.2(a) for the simulations of τLA. It has been divided in m non-controllable users and n controllable users. Each one of the n controllable users have the same neural controller as the one of Figure 5.2(b). The environment is governed by Equation 5.1 and it is necessary that τLA coordinates the n different users in order to smooth $s[k]$. In this case, for the evaluation of the τLA, only one input signal is used to simulate the non-controllable power profile.

The aggregated consumption of the m non-controllable individuals is now modelled as the aggregated consumption of a real grid. However, it is not exactly the same signal ($z_7[k]$), used in Section 4.5 and shown in Figure 4.3. A new signal has been synthesized which consists of a sum of sinusoids. Five frequencies have been selected, which are the ones with greater amplitude in the spectrum of Figure 4.3(b).

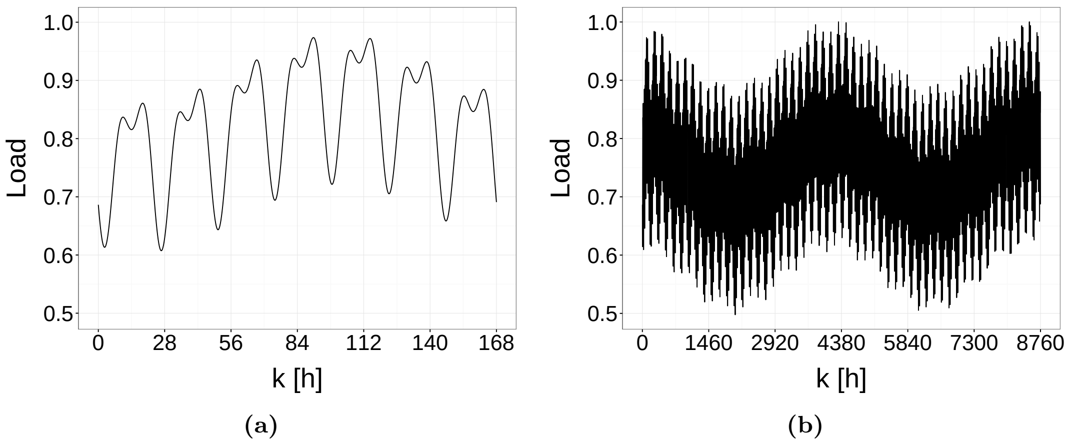


Figure 5.7: Non-controllable demand of sinusoidal form corresponding to the most significant frequencies of a real grid aggregated consumption. The frequencies selected are the half daily, daily, weekly, monthly and annually. (a) one week representation and (b) one year representation.

The frequency of each sinusoid corresponds to half daily, daily, weekly, monthly and annually frequencies. The mathematical form of $z[k]$ is presented in Equation 5.23.

$$\begin{aligned} z[k] = A_1 \cdot \cos(730 \cdot f + \varphi_1) + A_2 \cdot \cos(365 \cdot f + \varphi_2) + \\ + A_3 \cdot \cos(52 \cdot f + \varphi_3) + A_4 \cdot \cos(12 \cdot f + \varphi_4) + A_5 \cdot \cos(f + \varphi_5) + \mu \end{aligned} \tag{5.23}$$

where A_i is the amplitude of each sinusoid extracted from the spectrum of the grid from Figure 4.3(b), all frequencies have been normalized by the annually frequency and μ is the mean value of the signal taken from $f = 0$ of the grid spectrum.

The idea of using this type of signal is to get closer to a real scenario in which the controllable demand should be used to flatten the aggregated consumption. The resultant $z[k]$ can be seen in Figure 5.7, in which it can be observed the similarity with the form of the aggregated consumption of a grid. The aggregated sinusoidal grid consumption is normalized with respect to the number of individuals that form part of $z[k]$. Figure 5.7(a) shows the variability of the power profile during a week, in this case the pronounced difference between weekdays and weekends is not as high as in the real one. But there exists a considerable difference between the days of a week. Moreover, the annually aggregated consumption of Figure 5.7(b) shows the variability of the aggregated consumption along the whole year. The aggregated demand presents seasonal differences along the year and the consumption is always different from 0. Thus, this synthetic signal includes the most prominent behaviors of a real grid aggregated consumption. This is exactly what it is needed to test the coordination capabilities of τLA for an ensemble of controllable users.

On the other hand, the controllable part of the environment is formed by n identical evolved neural controllers. Thus, in order to test the algorithm, the controllable capacity of these n users is varied. The aggregated controllable demand, $x[k]$, is going to change its contribution to the environment, by increasing or decreasing its power to study the effect on the flattening of the demand. So, the controllable load capacity (L_C) is defined as

$$L_C[\%] = \frac{\max x[k]}{\max s[k]} \cdot 100 \tag{5.24}$$

where $\max x[k]$ is the maximum of the controllable aggregated consumption and $\max s[k]$ is the maximum of the total aggregated consumption. Hence, this ratio relates the amount of controllable power to the total. Moreover, with the definition of L_C, it is also known the amount of power that the non-controllable users represent.

For the simulations, an amount of power is set for the controllable users through L_C and at the same time, the n size of controllable users is varied. Therefore, the test will consist of analysing if the τLA is able to coordinate different ensembles of neural controllers and if their self-organization is able to reduce the variability of the non-controllable signal by obtaining a smooth $s[k]$. In the simulations, the percentage of L_C varies from 0 % to 100 %. With this sweep of L_C, it is analyzed the full range of situations that may occur in the environment, from an initial state at which there is no controllable demand (L_C near 0 %) until all the demand is controllable (L_C near 100 %). Per each L_C, the number of controllable users n is varied. The distribution of the number of individuals is not linear because it is necessary to test that τLA works for every population. Thus, it will be used a population set that grows exponentially whose values are $n \in \{1, 2, 5, 10, 25, 50, 100, 250, 500, 1000\}$. The idea of this set of individuals is to increase the difficulty of coordination for the algorithm. The reason is that for small populations the variations that individuals can introduce in the environment are smaller than the ones of bigger populations. So, simulations with few individuals are easier to solve than the ones with bigger populations since less instabilities are present and fewer individuals need to coordinate.

The different simulation elements have just been described but the evaluation method has not been discussed yet. This method will consist of analyzing the global behavior of the environment through the status of $s[k]$. For this purpose, 3 figures of merit are used: i) crest factor (C_f), ii) load factor (L_f) and iii) demand factor (D_f). In this case, all three parameters evaluate the form of the signal and assess its smoothness. Thus, the performance of the algorithm is based on how flattened is $s[k]$. The reason is that the coordination of all the individuals must reduce the signal variability. The first evaluation coefficient was already used in the post-evaluation of the evolved neural controller. On the other hand, the other two evaluation coefficients are used in a real grid context to evaluate its dimension with respect to the maximum load. These three parameters are as follows:

- *Crest factor (C_f).* It has been already defined in Section 4.5 and its mathematical expression is gathered in Equation 4.31.

- *Load factor (L_f).* The second evaluation coefficient is based on the use of the resources from the grid. It measures the average consumption in a period with respect to the maximum consumption. Thus, this coefficient serves as a measure to verify that the designed capacity of an electrical grid meets the demand. Equation 5.25 shows the mathematical expression of this ratio which consists of the average load divided by the peak load in a specified time period.

$$L_f = \left.\frac{\overline{s[k]}}{s_{peak}}\right|_K \tag{5.25}$$

  where, $\overline{s[k]}$ is the average environment signal in a period K and s_{peak} is the maximum of the environment signal in a given time period. The value of this ratio is less than or equal to 1. Typically, this coefficient has a value less than 1 because the facilities never operate at full capacity for the duration of the entire period selected. The interpretation of this coefficient is that the closer L_f is to 1, the more constant is the usage of the grid. On the other hand, low values of L_f show a demand profile in which occasionally high peaks appear. Thus, the best value of L_f is closer to 1 and indicates that $s[k]$ is flattened.

- *Demand factor (D_f).* This is the last factor of the evaluation and it also measures the use of the resources. D_f expresses the relationship of the amount of consumed power relative to the maximum power available that could be consumed in the grid. Thus, D_f compares the maximum demand in a time period with respect to maximum consumption of the system:

$$D_f = \left. \frac{\max s[k]}{s_{max}} \right|_K \tag{5.26}$$

where, $\max s[k]$ is the maximum of $s[k]$ in the specified time period K and s_{max} is the maximum possible load of the system. This is a measure of the occupancy of the system and it is relevant when trying to design the amount of load that a system should be rated for. This factor is always less than or equal to 1. The closer D_f is to 1, the greater the use of the system is. In this way, the system would not be oversized and the resources will be used constantly, having again a flattened consumption.

In the simulations, $z[k]$ has been sampled every 15 minutes. The total length of a simulation is 35040 samples, which is the time length of a year sampled every 15 minutes. In addition, the size of W_K has to be adjusted for the algorithm in order to evaluate the performance of the controller. The length of the chosen W_K is of a day, because during this period $s[k]$ presents various peaks and it is enough time for the ensemble of neural controllers to be coordinated and flattened the aggregated $s[k]$. Each experiment is repeated 30 times with different seeds to randomize the process and test if similar solutions can be obtained for different starting points. In addition, the sweep done in the L_C goes from 0 % to 100 % in steps of 5 % for the 10 sizes of individuals in the environment defined previously (from 1 to 1000). Then, 6300 experiments are simulated to test the algorithm capabilities.

In the evaluation process, three figures of merit are defined. However, it was not mentioned the evaluation time in which they are going to be evaluated. As observed in Figure 5.7, there exist different peaks in the grid signal. Thus, the EDeNC effectiveness is evaluated in order to cancel the main frequencies of the signal. The evaluation of each coefficient is going to be done by using the daily, weekly, monthly and annually periods. Hence, the controller behavior is tested during different time periods. The final result is the average of the number of times evaluated per time period, i.e.

$$\overline{C_{f,K}} = \frac{1}{N_K} \sum_{i \in N_K} C_{f,i} \tag{5.27a}$$

$$\overline{L_{f,K}} = \frac{1}{N_K} \sum_{i \in N_K} L_{f,i} \tag{5.27b}$$

$$\overline{D_{f,K}} = \frac{1}{N_K} \sum_{i \in N_K} D_{f,i} \tag{5.27c}$$

where $K \in \{day, week, month, year\}$ and N_K is the number of periods in which $s[k]$ have been evaluated. Then, the average behavior of the controller is obtained along a year for these three coefficients.

It has been explained all the different parameters and the evaluation assessment that take part in these simulations. However, simulations were carried out in two type of environments depending on how $s[k]$ is interpreted. The simulations have been divided in environment A (see Section 5.3.1) and environment B (see Section 5.3.2). In the first one, the maximum consumed power is limited while in the second one there will be no restriction. As in Chapter 4, all the simulations are done using the same hardware and software, *neuralSim*, already introduced. Thus, the results for each environment are as follows.

5.3.1 Environment A: power constraints

In this environment, it is considered that the maximum power that can be consumed by all individuals is limited. So, the controllable users have to adapt their consumption to the non-controllable demand in order not to consume more than the available maximum. With this environment, it is studied the adaptivity of the Evo-Devo Neural Controller (EDeNC), and the ability of the τ-Learning Algorithm (τLA) to coordinate

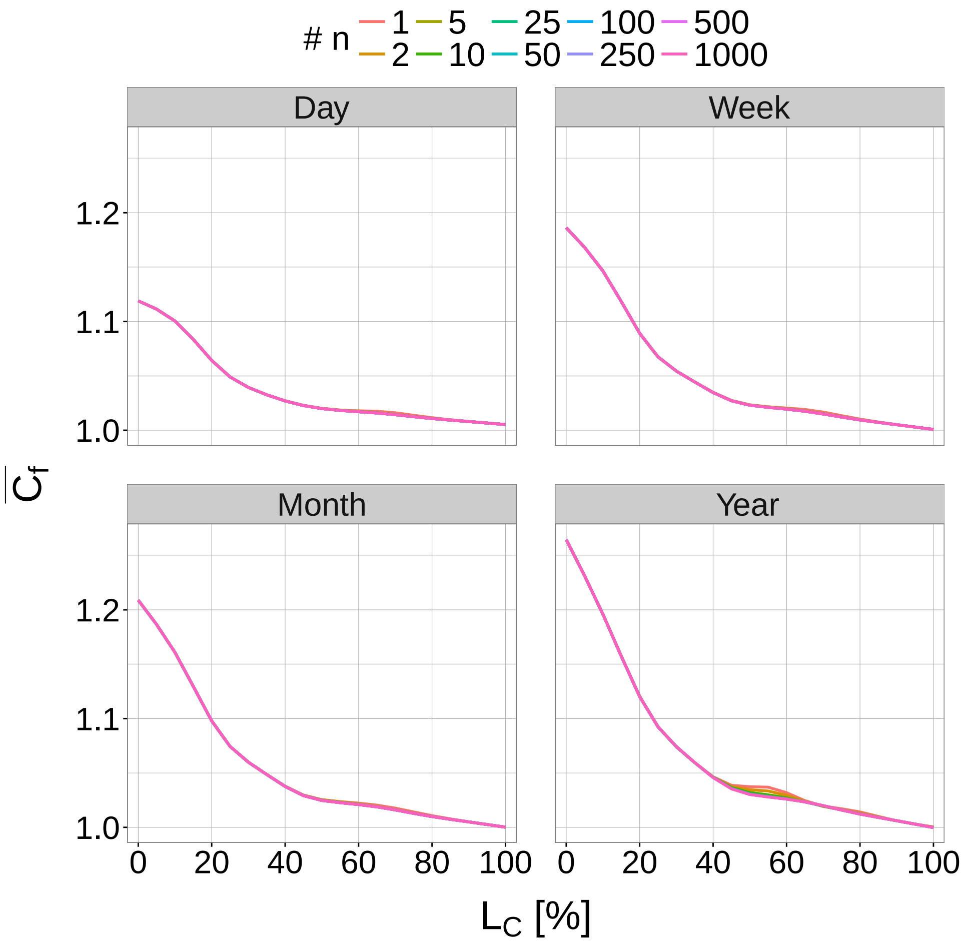


Figure 5.8: $\overline{C_f}$ for environment A evaluated for four different periods of time: daily, weekly, monthly and annually. The x-axis represents the variations of controllable capacity (L_C) and the color of each line represents a different size of controllable users ($\#n$).

different sizes of controllable ensembles for a given maximum power. This test wants to cope with the case in which an existing grid is already working and new users are included without modifying the total power capabilities of the network.

A microgrid is a good example of this behavior which consists of a small community with some sources of local energy generation. In this microgrid, the generation is designed in order to supply the local demand. Thus, if new users are added to this microgrid, the local energy source would not be able to meet the demand and it would be necessary to add more generation. However, if new users could adapt their consumption to the old ones, it would not be necessary to add more generation and the use of the system would be improved.

In this case, the maximum of $s[k]$ is limited to 1. For this purpose, the sum of the m non-controllable users and the n controllable ones must not exceed this limit. Therefore, $z[k]$ and $x_i[k]$ have been normalized to take values in the range $[0, 1]$. Moreover, under these conditions, there is no necessity to evolve again the best neural controller obtained in Chapter 4. Thus, the chromosome used in each neural controller for $x_i[k]$ is the one evolved previously.

The average crest factor, $\overline{C_f}$, results are shown in Figure 5.8. The results have been separated for the four different periods of time in which $s[k]$ is evaluated. Each point of the line is the median value of the $\overline{C_f}$ for the 30 seeds. In this case, the InterQuartile Range (IQR) is low, so the quartiles are very close and they cannot be seen. This means that τLA reaches similar solutions regardless of the seeds used for the environment configuration which implies that it is independent of the initial conditions of the simulation. It can be also observed that the differences between the number of controllable users are negligible for all the periods. There are slightly disparities between different population sizes when the evaluation period is high (annual period) and the amount of controllable load is the same as the non-controllable ($L_C \approx 50\%$). Hence, τLA is able to coordinate different ensembles of controllable users, independently of the size of the ensemble and the starting point of the algorithm.

In order to evaluate the effects of the controllable users, the improvement of the $\overline{C_f}$ is defined as

$$\nabla\overline{C_f}(L_C) = \frac{|\overline{C_f}(L_C) - \overline{C_f}(0\%)|}{\overline{C_f}(0\%)} \cdot 100 \tag{5.28}$$

In this case, it can be observed in all periods of evaluation an exponential decay towards the best value ($\overline{C_f} = 1$) with the increase of L_C. For all periods of time, any controllable capacity improves the network situation. In fact, a strong reduction of $\overline{C_f}$ can be observed in all the cases with relatively low fractions of controllable load. For example, for $L_C = 10\%$, relative reductions between 5 % and 10 % are achieved, compared to the base-case (non-controllable load). Then, the decay continues at a slightly reduced rate in the mid region of controllable load fraction ($L_C \approx 40\%$). The same happens for high factors of controllable load ($L_C > 50\%$), with the best value of $\overline{C_f} = 1$, reached for $L_C = 100\%$. Another observation is that the initial value of $\overline{C_f}$ for $L_C = 0\%$ grows according to the period of evaluation taken. The reason is that there is more variability of the signal for a whole year than in a day (see Figure 5.7). But in all the situations the EDeNCs are able to reduce the peaks of the environment signal towards a flattened $s[k]$. Note that τLA is able to coordinate all the controllable users in the environment even in the absence of non-controllable demand and is able to produce a constant $s[k]$ since $\overline{C_f} = 1$.

For the daily period, the value of $\overline{C_f}$ in absence of controllable users is equal to 1.119. Whereas, the use of the EDeNC is able to reduce it to 1.005 when all the demand is controllable. It is necessary that the controllable demand represents a 40 % of the total demand in order to obtain a considerable reduction of the $\overline{C_f}$ ($\nabla\overline{C_f} \approx 8\%$) with respect to the situation of non-controllable demand. The maximum reduction is achieved when $L_C = 100\%$, in which the $\nabla\overline{C_f} = 10.16\%$.

For the weekly period of evaluation, the initial value of $\overline{C_f}$ without controllable demand is 1.185, which is higher than the daily period of evaluation. The reason is that there is more variability in the waveform along a weekly period than in a day. In this case, from $L_C = 40\%$, the reduction of $\overline{C_f}$ is equal to $\nabla\overline{C_f}(40\%) = 12.65\%$. And the maximum reduction obtained is for a $L_C = 100\%$, $\nabla\overline{C_f}(100\%) = 15.55\%$, which corresponds to $\overline{C_f} = 1.0007$, being closer to 1 than in the daily period.

The next time period evaluated is the monthly one. In this case, the $\overline{C_f}$ without controllable load is higher than in the weekly and daily periods, and its value is 1.207. Again, the reason is the presence of more peaks during this evaluation period and the increase of variability. Now, for a controllable capacity of 40 % the decrease of the $\overline{C_f}$ is considerable, $\nabla\overline{C_f}(40\%) = 14.02\%$. This value represents near the double of the daily period evaluation, so that this figure indicates that the variability is highly reduced. The maximum reduction is achieved by $L_C = 100\%$. The $\overline{C_f}$ value is 1.0002 and the percentage of reduction compared to the case without controllable demand is $\nabla\overline{C_f}(100\%) = 17.17\%$.

Finally, for the last evaluation period, $\overline{C_f}$ with a $L_C = 0\%$ is the higher of all evaluation periods and it is equal to 1.263. In this case, for $L_C = 40\%$, the $\overline{C_f}$ reduction achieved is $\nabla\overline{C_f}(40\%) = 17.13\%$, which means a great reduction in the

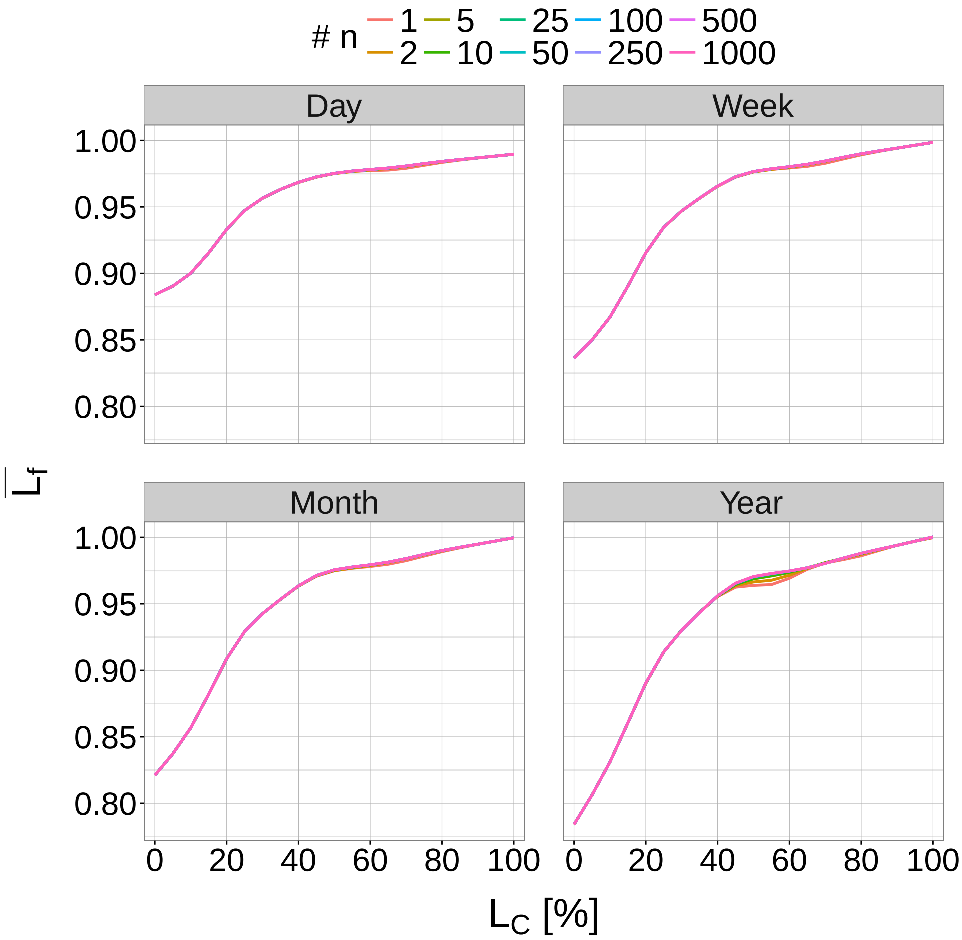


Figure 5.9: $\overline{L_f}$ for environment A evaluated for four different periods of time: daily, weekly, monthly and annually. The x-axis represents the variations of controllable capacity (L_C) and the color of each line represents a different size of controllable users ($\#n$).

peaks of the waveform towards a constant $s[k]$. The reduction achieved for $L_C = 100\%$ has a value of $\nabla\overline{C_f}(100\%) = 20.85\%$ and the final $\overline{C_f}$ value is 1.0003. From these results, the EDeNCs achieve the goal of reducing the peaks of the environment signal, obtaining favorable results for all the controllable load percentage. In Table 5.1, all the results for the different percentage of L_C and periods of evaluation are gathered.

The EDeNC evaluation continues by analysing the next coefficient of the process. So, the second factor of the assessment is the average load factor per period of evaluation, $\overline{L_f}$, which will give us information about how fitted is the grid with respect to the demand through the measure of the system resources' occupation. Figure 5.9 collects all the simulation results for this coefficient. In order to compare the results of Figure 5.9, they have been divided in 4 panels, one per each period of evaluation. As the results of $\overline{C_f}$ of Figure 5.8, each point of the lines of Figure 5.9 represents the median of each evaluation period for the 30 seeds. There is almost no dispersion and all the quartiles are really close to each other. The reason is that the development of the evolved neural controller through the τLA reaches very similar solutions for all the 30 different starting points. It is also confirmed, as in the analysis of the $\overline{C_f}$, that the algorithm is able to coordinate the different ensembles of neural controllers independently of the number of controllable users. Although it obtains similar solutions for all of them, the output of each user is not the same. τLA is able to adjust the output of each controllable user depending on the number of individuals and L_C.

Figure 5.9 shows a logarithmic growth of $\overline{L_f}$ for all periods of evaluation. The bigger the evaluation period is, the lower $\overline{L_f}$ ($L_C = 0\%$) is. This behavior is due to the peaks and variability of the $z[k]$ signal, which are higher when the evaluation period is larger. As mentioned above, the difference in the results between the number of controllable users are negligible. However, little differences can be observed for the annual evaluation period and a controllable capacity around $L_C \approx 50\%$, in which the value of $\overline{L_f}$ is lower for large sizes of controllable users. A strong increase of the $\overline{L_f}$ can be observed in all cases with relatively low fractions of controllable loads. Moreover, the growth slows down from a $L_C \geq 40\%$ in all cases. The closer to 1 $\overline{L_f}$ is, the greater the flattening of $s[k]$ is. Note that the $\overline{L_f}$ maximum value is almost reached for $L_C = 100\%$ and also τLA is able to coordinate the ensemble of users in the absence of $z[k]$ by achieving a flattened response. In all cases, the EDeNCs are able to improve the average use of the available resources since $\overline{L_f}$ positive grows for all L_C values.

In the case of an evaluation period of a day, the value of $\overline{L_f}$ in the absence of controllable demand is 0.8838. This value is increased until it reaches a maximum of 0.9896 in the best case, when the entire environment consists only of controllable users. In order to measure the improvement of $\overline{L_f}$ for the different L_C values, $\Delta\overline{L_f}(L_C)$ is defined as

$$\Delta\overline{L_f}(L_C) = \frac{|\overline{L_f}(L_C) - \overline{L_f}(0\%)|}{\overline{L_f}(0\%)} \cdot 100 \tag{5.29}$$

At the point where the change of slope occurs ($L_C = 40\%$), the improvement of $\overline{L_f}$ is $\Delta\overline{L_f}(40\%) = 9.53\%$. Whereas the maximum $\overline{L_f}$ is reached at $L_C = 100\%$ and the percentage of this improvement has the value $\Delta\overline{L_f}(100\%) = 11.97\%$. For this evaluation factor, the difference between the case of $L_C = 40\%$ and the total controllable demand is not as pronounced as for $\overline{C_f}$. The reason is that the average demand takes advantage of almost all the available resources, although peaks still exist in the waveform of $s[k]$.

For the weekly evaluation period, $\overline{L_f}$ starts with a value of 0.8371 for the case in which non-controllable demand is set in the environment. This parameter presents a lower value than the previous evaluation period because the average load decreases due to higher variability, therefore $\overline{L_f}$ decreases. In the change of tendency point, it is obtained that $\overline{L_f}$ has improved in $\Delta\overline{L_f}(40\%) = 15.29\%$. For the maximum controllable capacity, $\overline{L_f} = 100\%$, the improvement obtained is $\Delta\overline{L_f}(100\%) = 19.28\%$, which corresponds to $\overline{L_f} = 0.9985$.

For the monthly evaluation period, $\overline{L_f}$ has a value for $L_C = 0\%$ lower than the previous two evaluation periods and it has a value of 0.8219. This value is similar to the weekly evaluation period because the average demand in a month is similar to the average of the weeks inside the month. For $L_C = 40\%$, $\Delta\overline{L_f}(40\%) = 17.14\%$, which is also pretty similar to the value of the weekly evaluation period. And for $L_C = 100\%$, $\overline{L_f} = 0.9996$ which meant an improvement of $\Delta\overline{L_f}(100\%) = 21.62\%$ with respect to the absence of controllable demand.

The annual $\overline{L_f}$ for $L_C = 0\%$ has a value of 0.7850, which is also the minimum value of all the evaluation periods. The reason is that the whole simulated environment signal is taken into account and there is a significant variability that makes difficult to use the entire available resources. $\overline{L_f}$ grows faster until it reaches $L_C = 40\%$, which represents an improvement of $\Delta\overline{L_f}(40\%) = 21.67\%$. And then, the $\overline{L_f}$ continues growing slowly until it reaches a value of $\overline{L_f} = 0.9998$, an improvement of $\Delta\overline{L_f}(100\%) = 27.42\%$ with respect to the non existence of controllable demand.

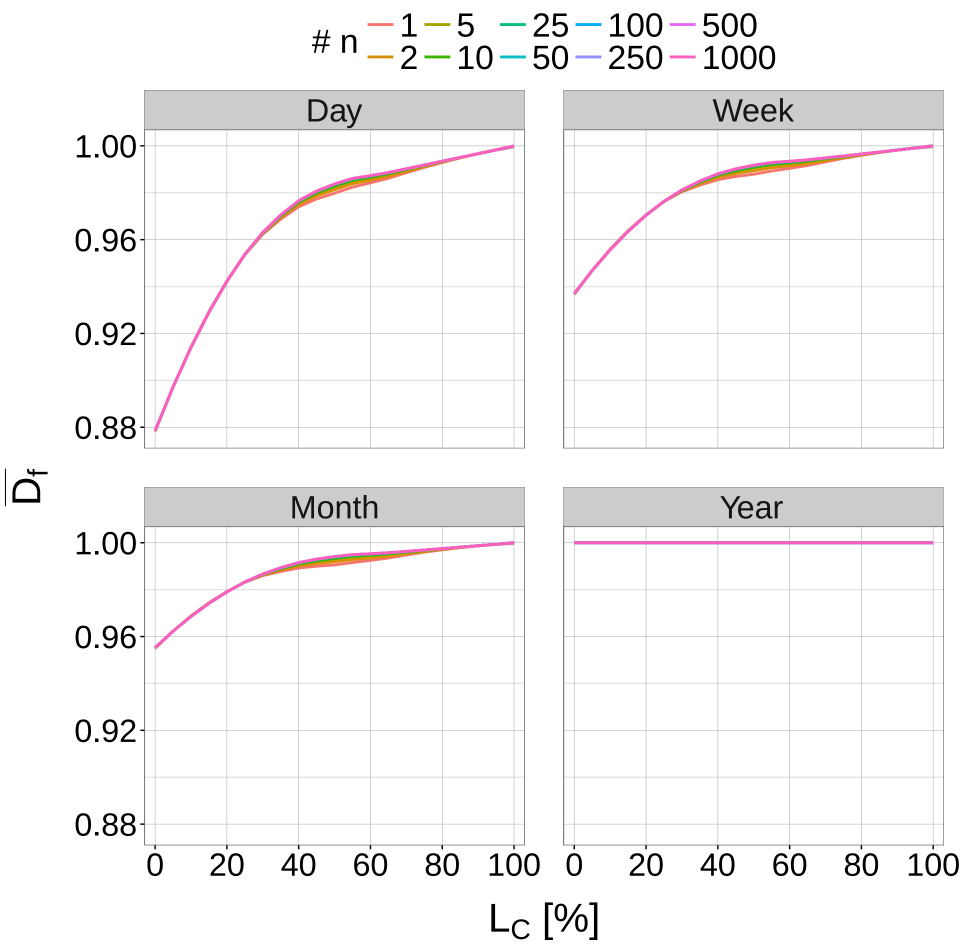


Figure 5.10: $\overline{D_f}$ for environment A evaluated for four different periods of time: daily, weekly, monthly and annually. The x-axis represents the variations of controllable capacity (L_C) and the color of each line represents a different size of controllable users ($\#n$).

All the results for the different $\Delta\overline{L_f}$ of all the evaluation periods and L_C are gathered in Table 5.1.

In Figure 5.10, $\overline{D_f}$ results are gathered for the different configurations. The results have been divided in 4 panels one per period. As in previous assessments, it is represented the median and standard deviation of the $\overline{D_f}$ per period of evaluation and configuration of the environment. The deviation of the 30 seeds is low since the IQR is low and the quartiles are very close and they cannot be seen. τLA reaches similar solutions for all the different configurations regardless of the seed used and the differences between the number of controllable users are negligible for all the periods. This behavior continues supporting that the τLA effectiveness is independent from the starting point of the simulation.

Note that in the case where the evaluation period is one year, the results for all the sizes and L_C fractions are the same because the maximum in this period is the same amount for numerator and denominator of the coefficient. In this case, it also presents a little more variability for different sizes of controllable users. For all periods except the annual, all of them present a difference in the result less than 1% between a 35% and 65% L_C fraction for the different sizes of controllable users.

Figure 5.10 shows a logarithmic growth as the results of $\overline{L_f}$ for daily, weekly and monthly results. In this case, unlike the $\overline{L_f}$ results, the smaller the evaluation period is, the bigger the value of $\overline{D_f}$ for a $L_C = 0\%$ is. This behavior is because the consumption of most of the days is far from the maximum annual consumption. In addition, it can be observed a rapid growth for $L_C < 40\%$. This growth slows down until it reaches its maximum for $L_C = 100\%$. The best value for all the cases is 1, which means that the system is using all the resources during the entire evaluation period. With the EDeNCs, the system efficiency is increasing by using its full capacity and producing a flat signal that takes advantage of it.

For a daily evaluation period, $\overline{D_f}$ is 0.876 for $L_C = 0\%$. Then, as the controllable capacity increases in the environment this value grows until it reaches its maximum ($\overline{D_f} = 1$) for $L_C = 100\%$. As done before, a factor is defined that measures the improvement of adding more L_C. This coefficient has the following form,

$$\Delta\overline{D_f}(L_C) = \frac{|\overline{D_f}(L_C) - \overline{D_f}(0\%)|}{\overline{D_f}(0\%)} \cdot 100 \tag{5.30}$$

In the point at which the speed of growing reduces, the improvement achieved is $\Delta\overline{D_f}(40\%) = 11.05\%$. Whereas for the maximum value of the factor, $s[k]$ was improved in $\Delta\overline{D_f}(100\%) = 13.72\%$. At $L_C = 100\%$, $\overline{D_f}$ has reached its maximum value so the $\Delta\overline{D_f}(100\%)$ is the maximum value that $s[k]$ could improve with the help of the EDeNC.

The next evaluation period is the weekly period. In this case, $\overline{D_f}$ begins with a value of 0.936 for $L_C = 0\%$. It is bigger than the daily period since the peaks during the week are higher than for one day. At a $L_C = 40\%$, the EDeNC have improved the factor in $\Delta\overline{D_f}(40\%) = 5.56\%$, which is lower than the previous value since it starts closer to the maximum value. With $L_C = 100\%$, it is obtained a $\overline{D_f} = 1$ with an improvement of $\Delta\overline{D_f}(100\%) = 6.82\%$ with respect to the case with no controllable load. As can be observed, the improvement for the best L_C is similar to a $L_C = 40\%$. The reason is that the maximum peak of the week is closer to the maximum of the system, but the algorithm continuous improving this value until its maximum is reached.

The last period analysed is the monthly period, since the improvement of the annual period is inexistent as it always uses the maximum capacity of the system. A bigger value of $\overline{D_f}$ is achieved for $L_C = 0\%$ which is equal to 0.955. It also reaches the maximum value of $\overline{D_f} = 1$ for $L_C = 100\%$, which means a maximum improvement of $\Delta\overline{D_f}(100\%) = 4.76\%$ (lower than the previous periods due to being close to the maximum). Meanwhile, for $L_C = 40\%$, the algorithm achieves $\Delta\overline{D_f}(40\%) = 3.89\%$ which is closed to the maximum possible value of improvement. By following the same reasoning as with the weekly period, the maximum peak found during the evaluation of a month is closer to maximum of the system, so the $\overline{D_f}$ is closer to 1. The rest of the results are summarized in Table 5.1 for all the periods and L_C.

One last result that it is going to be analyzed is how flat the environment signal is for the different controllable fractions, L_C. In order to do this analysis, the representation of the load duration curve explained in Section 2.1.5 is used. This representation consists of sorting decreasingly the consumption along the 8760 h of a year. Thus, the maximum consumption of the year is at 0 h and the minimum is located at 8759 h. And in order to compare them, the load is normalized by the maximum load available in the environment, i.e. the sum of the maximum load that each user can consume. Figure 5.11 shows the different load duration curves for different controllable fractions.

In view of these results, it can be concluded that the use of any controllable demand will improve the grid status since the variability of $s[k]$ is reduced (the value of the slope decreases) and also the maximum consumption of the grid is reduced as L_C grows. Another finding that emerges from these results is that the demand

L_C[%]	Day			Week			Month			Year		
	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$
0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	1.67	1.87	3.96	3.26	3.59	2.24	3.90	4.27	1.58	5.37	5.93	0.00
20	4.87	5.54	7.37	8.06	9.30	3.71	9.04	10.48	2.60	11.26	13.35	0.00
30	7.15	8.28	9.48	11.06	13.19	4.78	12.27	14.75	3.34	15.00	18.57	0.00
40	**8.19**	**9.53**	**11.05**	**12.65**	**15.29**	**5.56**	**14.02**	**17.14**	**3.89**	**17.13**	**21.67**	**0.00**
50	8.93	10.44	11.83	13.79	16.83	5.77	15.26	18.86	3.96	18.37	23.54	0.00
60	9.08	10.63	12.10	13.95	17.04	6.01	15.41	19.08	4.20	18.71	24.05	0.00
70	9.35	10.96	12.53	14.36	17.61	6.23	15.86	19.73	4.35	19.26	24.91	0.00
80	9.66	11.34	12.96	14.80	18.23	6.44	16.35	20.42	4.50	19.84	25.81	0.00
90	9.93	11.67	13.35	15.20	18.78	6.63	16.78	21.05	4.63	20.35	26.63	0.00
100	**10.16**	**11.97**	**13.72**	**15.55**	**19.28**	**6.82**	**17.17**	**21.62**	**4.76**	**20.85**	**27.42**	**0.00**

Table 5.1: Comparison of $\overline{C_f}$, $\overline{L_f}$ and $\overline{D_f}$ for all the periods of time and all controllable capacities in environment A. In bold, the results discussed throughout the text are highlighted.

curve gets flatter as L_C grows and the slope of the load duration curve decreases. Regardless the size of the environment, the coordination of the different users always improves its status. Furthermore, note the fast improvements introduced by relatively low values of the controllable fraction ($L_C \leq 20\%$). It is interesting to analyze the first 10 h of higher demand in which it can observed an important reduction of the system marginal costs in order to meet the maximum power demand. For $L_C = 40\%$ the slope of the load duration curve has been reduced approximately in an 80 % with respect to the case with no controllable demand. The maximum has been reduced around a 25 %. Whereas for $L_C = 100\%$ the slope reduction is of 100 % and the maximum reduction is around 50 %. In conclusion, the EDeNCs are able to improve the state of $s[k]$ and make a better use of the available resource as observed in the results presented in this Section.

5.3.2 Environment B: no power constraints

Significant results have been achieved for bounded environments, in which the EDeNCs must adapt their output in order to cancel the variability of $s[k]$. However, in order to get closer to a real grid environment, the consumed power cannot be constrained. The reason is that grids are constantly evolving and their boundaries grow as the needs of their users also increase. In this Section, it is considered an environment in which the maximum generation capacity of the system is bigger than the maximum power that the different users can consume. Thus, there are no power constraints and the users can consume what they need in order to get a smooth environment signal. This environment configuration analyzes that the controllable users are able to self-organize with the non-controllable ones and achieve a flattened $s[k]$ regardless any limitation to their consumption.

As in Section 5.3.1, τLA is necessary to coordinate the different sizes of controllable ensembles and to guarantee the stability of $s[k]$ independently to the number of EDeNCs. The idea behind this environment is the design of a grid in which the controllable users will try to adapt their consumption to enhance the operation of the grid and relax the designing conditions of the system. In this case, the grid assures to meet the demand of the users at any time, so the power constraints disappear. In addition, the EDeNCs consume any amount of power, so the output is not limited to 1, which is the maximum output of the EDeNC by design. Each $x_i[k]$ is denormalized multiplying it by a maximum power ($p_{x,max}$), being the new output range per controllable user $x_i[k] \in [0, p_{x,max}]$. Thus, it is not assured that $s[k]$ would be in the range at which the EDeNCs produces an antiphase output and it could be

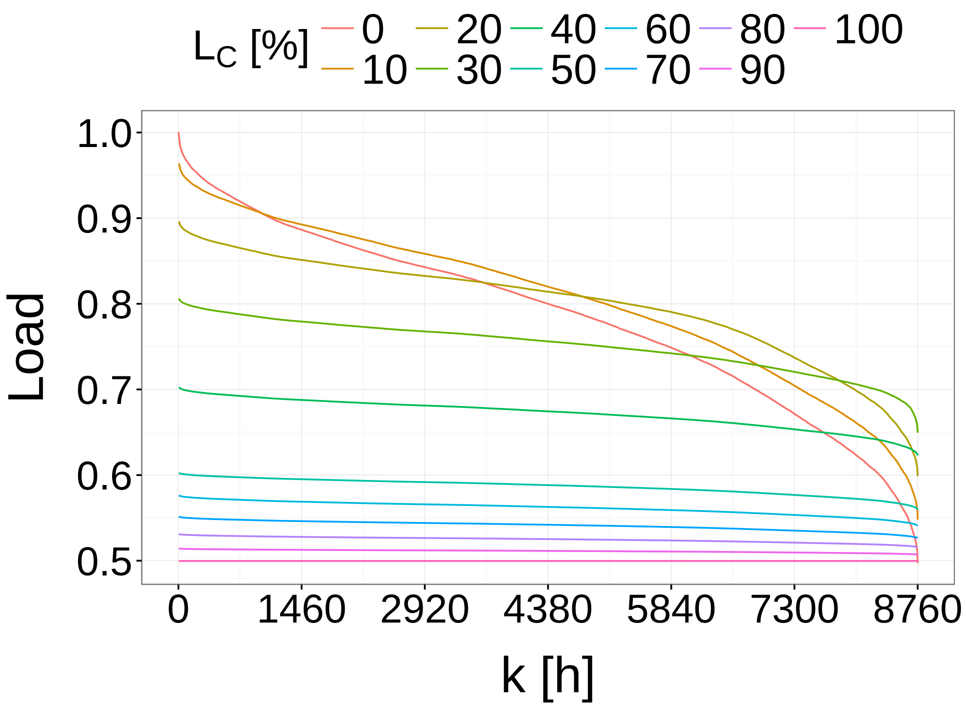


Figure 5.11: Load duration curve of environment A for different controllable load capacities.

in saturation or in inhibition regime. Therefore, it is necessary to add a preprocessing part which assures that $s[k]$ is within the firing range of the neural controller. To this extent, the environment signal is normalized with respect to the configuration of the environment by following Equation 5.31.

$$\hat{s}[k] = \frac{s[k]}{z_{max} + x_{max}} \tag{5.31}$$

where z_{max} is the maximum amplitude for the m non-controllable users and x_{max} is the maximum amplitude for the n controllable users. Thus, the environment is normalized by the maximum power that the users can consume.

$\hat{s}[k]$ will be at the firing range of the neural controller, even for the worst case in which all users consume at full capacity. Now, the input to each neural controller is normalized and belongs to $\hat{s}[k] \in [0, 1]$, guaranteeing that the neural controller always responds to it. This normalization of $s[k]$ implies that the EDeNCs could work under any combination of environment parameters. Furthermore, $\hat{s}[k]$ will provide the necessary information of the waveform to the controllable users since the information about the total capacity of the system is not revealed. In a real scenario, the normalized information will be provided to the users by the grid operators which have access to all the system information. So, this mechanism increases the security of the system and provides only the necessary information. This makes the system more robust and increases the efficiency of the shared information.

In this environment, all the controllable users have a $p_{x,max} = 1$ in order to simplify the environment calculus. Then, the maximum amplitude of the m non-controllable users is adjusted through L_C, i.e. $p_{z,max} = n \cdot p_{x,max} \cdot L_C$. This amount of power will be used to multiply it by the profile of Figure 5.7. So, each environment configuration will have a different amplitude of $s[k]$. In this case, the environment conditions have changed and the best evolved chromosome can no longer be used. The neural controller has been evolved again based on the best parameter scenario of

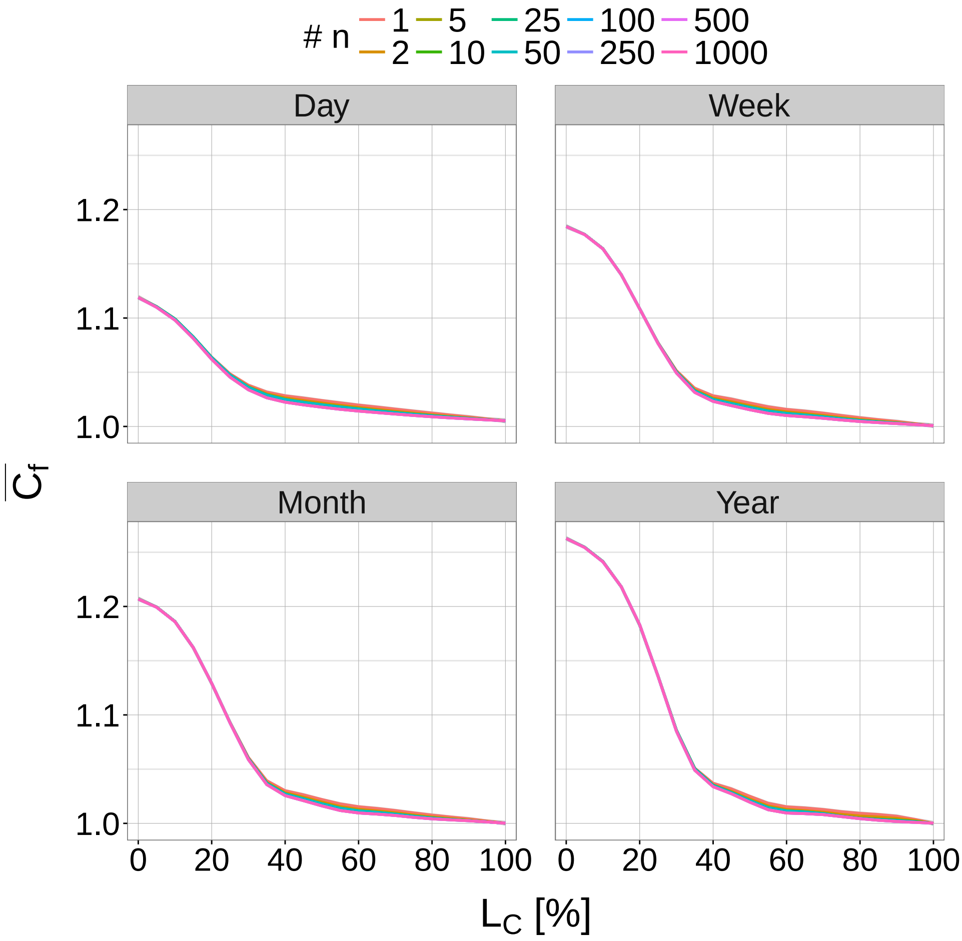


Figure 5.12: $\overline{C_f}$ for environment B evaluated for four different periods of time: daily, weekly, monthly and annually. The x-axis represents the variations of controllable capacity (L_C) and the color of each line represents a different size of controllable users ($\#n$).

Chapter 4 with the new environment conditions. After the evolution, the new best chromosome is used in each EDeNC to test this new environment closer to the reality.

The first factor to analyze is $\overline{C_f}$ and all results are gathered in Figure 5.12. These results are presented differentiating the evaluation period, the population size of the controllable users and the controllable fraction, L_C. As in environment A, each point of the representation is the median of the 30 seeds used to evaluate each configuration of the environment. The deviation per configuration is represented in shadow grey, but the $1st$ and $3rd$ quartile are too close to the median so it is negligible. Again, τLA develops the ensemble of evolved neural controllers in order to coordinate them and reduce the peaks of the aggregated waveform. In spite of obtaining similar results for all the parameter configurations, more discrepancies can be observed depending on the ensemble size for all periods. Best results are always achieved by bigger sizes of ensembles since τLA has major quantities of energy to adapt the controllable demand to the non-controllable one. The reason of this behavior is that for different L_C and n, $s[k]$ is different for each one and $\hat{s}[k]$ may differ for each one, so the solution might differ depending on the $\hat{s}[k]$. Despite these differences, the algorithm is able to coordinate the ensemble of controllers no matter the size and the starting point of the simulation.

At the view of these results, it can be concluded that the inclusion of controllable demand in the environment improves the waveform of the environment signal, making it more constant as L_C grows. This appreciation is observed for any of the evaluation periods since the $\overline{C_f}$ is always reduced. In Figure 5.12, an exponential decay is observed for all evaluation periods towards $\overline{C_f} = 1$ as L_C increases. It is also observed that for $L_C \leq 10\%$, the decay is slow. Then, it changes rapidly until it reaches a $L_C = 35\%$. And again the decay slows down until the best value of $\overline{C_f}$ is achieved for $L_C = 100\%$. In this case, the point at which the decay changes significantly is achieved before than in environment A. This is a positive effect because it indicates that the greatest benefits are obtained at low L_C, i.e. moderate investments. Moreover, the $\overline{C_f}$ starting point grows with the evaluation period, since the variability of the signal also increases. As in environment A, τLA is able to coordinate the EDeNC ensemble even in the absence of $z[k]$ and achieves a $\overline{C_f} = 1$.

In order to detail the performance of the EDeNCs, the results of Figure 5.12 are described as follows. For the case with $L_C = 0\%$ in the daily period, the $\overline{C_f}$ value is equal to 1.119, which is the same value as in environment A. The reason is that $\overline{C_f}$ is a waveform factor and it is independent of the scale. The application of τLA for the ensemble coordination is able to obtain a $\overline{C_f} = 1.005$, which is also the same value as in environment A. So, Equation 5.28 will give the same amount of improvement of $\overline{C_f}$, $\nabla\overline{C_f}(100\%) = 10.16\%$. However, at the point of change of slope $L_C = 35\%$, a value of $\nabla\overline{C_f}(35\%) = 8.07\%$ is obtained which is greater than in Environment A under the same percentage of controllability percentage. Therefore in both cases, the algorithm has developed the evolved neural controller to the minimum $\overline{C_f}$ possible.

The same behavior can be observed for a weekly period of evaluation. At the point $L_C = 0\%$, the $\overline{C_f}$ obtained has a value of 1.185. Then, for $L_C = 35\%$, the improvement of $\overline{C_f}$ is $\nabla\overline{C_f}(35\%) = 12.71\%$. $s[k]$ is almost flattened at this point, it only needs a small percentage to get its response as flatten as possible. Finally, the maximum reduction of $\overline{C_f}$ is achieved for $L_C = 100\%$ and has a value of 1.001. This value means an improvement of the waveform of $\nabla\overline{C_f}(100\%) = 15.55\%$.

The next period evaluated is the monthly one. At this period, the starting point of $\overline{C_f}$ is bigger than the previous ones because the variability of the signal is also increased since more peaks of $s[k]$ are evaluated. The value of $\overline{C_f}$ for $L_C = 0\%$ is 1.208. With the increase of L_C, this value is reduced and at a controllable capacity of 35 %, this reduction has a value of $\nabla\overline{C_f}(35\%) = 13.92\%$. The maximum reduction is achieved by $L_C = 100\%$, the value of the $\overline{C_f}$ is 1.0002 and the reduction of the $\overline{C_f}$ is of $\nabla\overline{C_f}(100\%) = 17.17\%$.

Finally, the last evaluation of the environment is done by taking into account the annual period signal. For this period, $\overline{C_f}$ has the biggest value due to the variability of the signal. At a $L_C = 0\%$, the $\overline{C_f}$ has a value of 1.263. Then, $\overline{C_f}$ is reduced as the controllability of the system grows. So, for the point which the slope changes its speed, $L_C = 35\%$, an improvement of $\nabla\overline{C_f}(35\%) = 16.52\%$ is achieved. For this value of L_C, the $s[k]$ has smoothed almost all the peaks presented in its waveform. The reduction of $\overline{C_f}$ continues growing until it reaches a value of 1.0001, which means an improvement of $\nabla\overline{C_f}(100\%) = 20.82\%$. For all the periods, the results at $L_C = 100\%$ are very similar, meaning that the coordination algorithm found the best possible solution to flatten $s[k]$. All the reduction results are gathered in Table 5.2 to see in more detail the values obtained for each controllability fraction.

The next evaluation coefficient is $\overline{L_f}$ which will report information about how good the average occupation of the environment is with respect to the maximum consumption. It is a measure of how well the grid meets its demand. Figure 5.13 shows all the result for the assessment of this factor. Note that, the closer $\overline{L_f}$ is to 1, the more constant is $s[k]$. The four evaluation periods are represented in a different panel of Figure 5.13. All the points are the median representation of evaluating

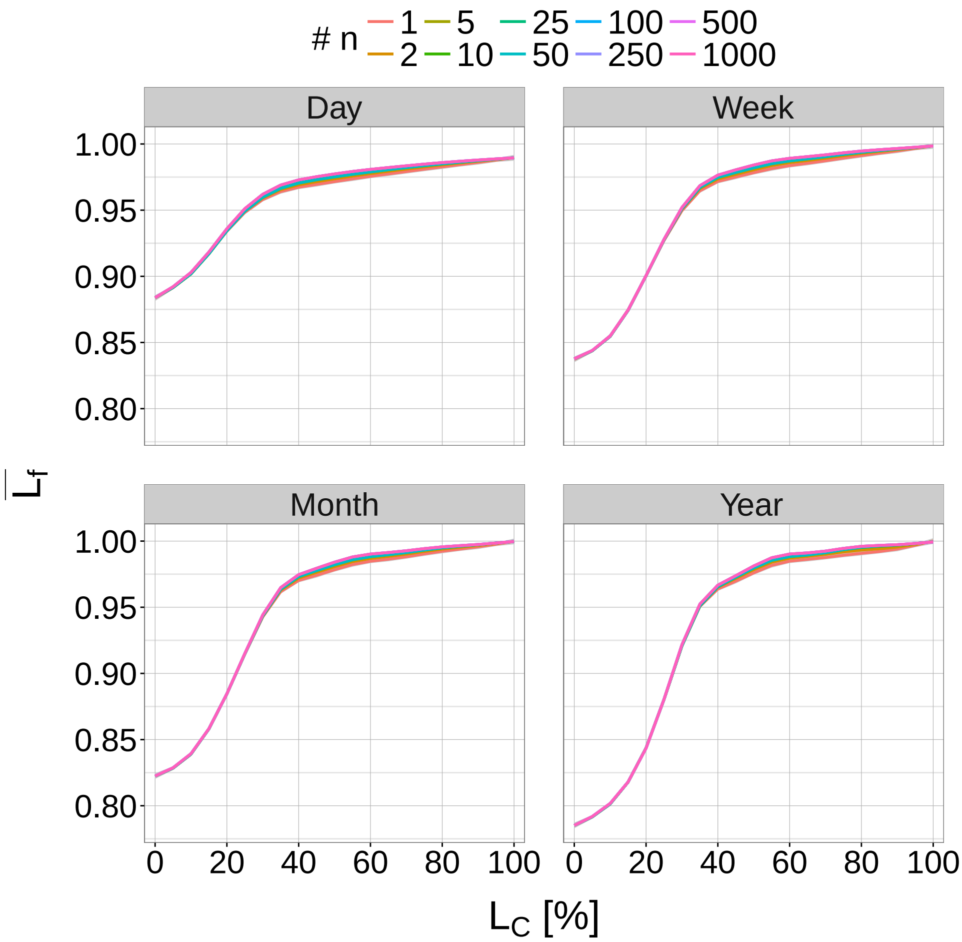


Figure 5.13: $\overline{L_f}$ for environment B evaluated for four different periods of time: daily, weekly, monthly and annually. The x-axis represents the variations of controllable capacity (L_C) and the color of each line represents a different size of controllable users ($\#n$).

$\overline{L_f}$ in the corresponding time period for the 30 different seeds. In addition, the $1st$ and $3rd$ quartile are also represented as the limits of a grey shadow behind the line. However, τLA reaches a similar solution to all the 30 seeds so there is no deviation in the data. The algorithm is able to coordinate the ensemble of controllers for any size as the tendency of each line is positive and the value of $\overline{L_f}$ grows. However, more discrepancies are shown than in environment A due to the different $s[k]$ signals. Despite these small differences, the best results are achieved by bigger sizes of populations since the algorithm is able to displace major quantities of signal. τLA achieves similar results for all the 10 different sizes of controllable users in the environment and the controllable capacity.

For all the periods, the tendency of the $\overline{L_f}$ growth is logarithmic (see Figure 5.13). The value of $\overline{L_f}$ for a $L_C = 0\%$ is lower as the evaluation period increases its value. The reason is the disparity of the average load and the maximum of the environment as the period of evaluation grows for $z[k]$. It is also observed that for $L_C \geq 10\%$, $\overline{L_f}$ grows slowly, then it speeds up until $L_C \geq 35\%$ and again it slows down until $L_C = 100\%$. Even for the worst situation in which $z[k] = 0$, the algorithm is able to coordinate the ensemble of users achieving a value very close to the unity.

In order to insist about the EDeNC performance, the results per each period of evaluation are described. Again, all the values of the $\overline{L_f}$ for a $L_C = 0\%$ are the same, since the scale factor is not taken into account by this factor. In this case, for a daily evaluation period, $\overline{L_f}$ in the absence of controllable demand is 0.884. At the point at which the growing tendency change their value, the improvement obtained using Equation 5.29 is of $\Delta\overline{L_f}(35\%) = 9.40\%$. And the maximum value of $\overline{L_f}$ is 0.989 reached at $L_C = 100\%$ which means an improvement of $\Delta\overline{L_f}(100\%) = 11.97\%$. The difference in the results are not as pronounced as for $\overline{C_f}$, because the match of the average load with the capacity of the system is greater in spite of presenting peaks.

For the case of weekly evaluation periods, it follows the tendency as for daily ones. In this case, $\overline{L_f}$ starts with a value of 0.837 in the absence of controllable demand. Then, for the elbow point of the logarithmic growth ($L_C = 35\%$), the factor has increased $\Delta\overline{L_f}(35\%) = 15.38\%$. And for the best value of simulations, $L_C = 100\%$, it reaches a value of $\overline{L_f} = 0.998$ and an improvement of $\Delta\overline{L_f}(100\%) = 19.28\%$. This value is higher than for the daily period since it is easy to get closer to the system maximum for higher periods.

For a monthly period of evaluation, the starting value of the factor has decreased to $\overline{L_f} = 0.821$ in the absence of controllable demand. Then, this value is improved rapidly until it reaches $L_C = 35\%$ at which the improvement measured is $\Delta\overline{L_f}(35\%) = 17.00\%$. The maximum improvement is $\Delta\overline{L_f}(100\%) = 21.62\%$ for $L_C = 100\%$ and the value that the factor reaches is $\overline{L_f} = 0.9991$.

Finally, the last results for the entire signal evaluation length are as follows. The value of the factor for $L_C = 0\%$ is 0.785, which is the minimum value reached by all the evaluation periods. This value grows quickly until it reaches $L_C = 35\%$ at which the factor has already improved $\Delta\overline{L_f}(35\%) = 20.78\%$. $\overline{L_f}$ continues growing but at a speed slower than the previous one, obtaining an improvement of the factor for the maximum value of $\Delta\overline{L_f}(100\%) = 27.33\%$. At this point, the maximum value of $\overline{L_f}$ reached is equal to 0.9995. Again for this factor, as in the case of $\overline{C_f}$, the results at $L_C = 100\%$ are very similar for the two environments. The reason is that τLA reaches the best possible coordination at this point, because differences can be observed in the growing of the coefficient. All the different $\Delta\overline{L_f}$ for different controllable percentages are gathered in Table 5.2 in order to analyze them in more detail.

The last factor of the evaluation is represented in Figure 5.14. As the previous factors, $\overline{D_f}$ analyses the performance of the EDeNCs in the environment through the utilization of the maximum capacity of the system. The results are divided in 4 panels for each evaluation period of the factor, varying L_C and n. Each point, contained in the lines of Figure 5.14, represents the median of the average factor for the corresponding evaluation period. The same configuration of the simulation is repeated for 30 seeds, but as for previous results the deviation is so small that it is difficult to appreciate in Figure 5.14.

The results of the annual period of evaluation are always the same, since it is comparing the maximum reached for the signal length within the evaluation period with the maximum capacity of the system which in this case is the same. The rest of evaluation periods present a logarithmic growth. In this case, the smaller the evaluation period is, the bigger the starting value of $\overline{D_f}$ is. The reason is that for shorter evaluation periods its maximum consumption is far from the annual maximum. In this case, the change of tendency in the curve is produced for $L_C = 50\%$. They have similar results for all sizes of n. But it can be observed that for bigger values of n the value of $\overline{D_f}$ decreases a little as the L_C grows. The reason is that some disturbances occur during the coordination of the biggest ensembles for high L_C that maintain $s[k]$ flattened but its maximum value decreases. In spite of this behavior, the maximum value reached is close to 1, which means that the system is working to its maximum capacity the entire evaluation period. Therefore, with the EDeNCs, a smoothed $s[k]$ is produced that is enhancing the system as the working point is situated at the full environment capacity.

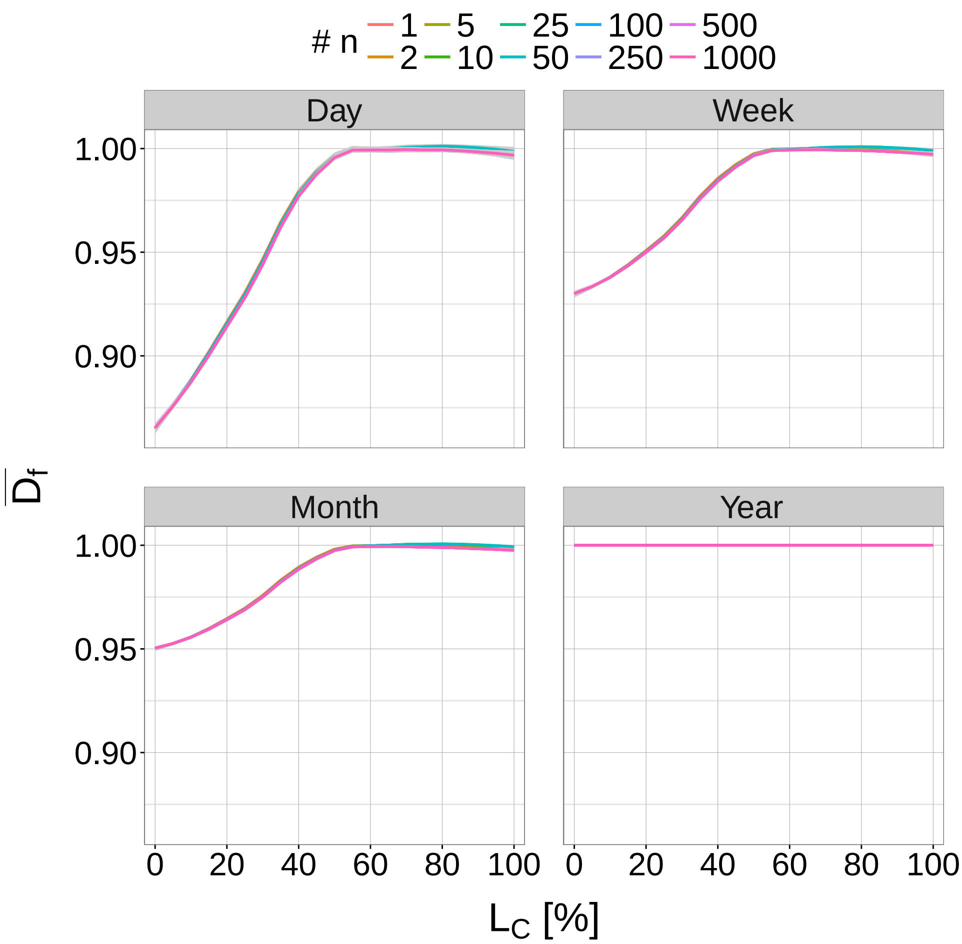


Figure 5.14: $\overline{D_f}$ for environment B evaluated for four different periods of time: daily, weekly, monthly and annually. The x-axis represents the variations of controllable capacity (L_C) and the color of each line represents a different size of controllable users ($\#n$).

The value of $\overline{D_f}$ for a daily evaluation period at the point of absence of controllable demand is 0.876. In this case, the improvement coefficient defined in Equation 5.30 is also used. Then, the value of $\overline{D_f}$ grows rapidly until it reaches $L_C = 50\%$. At this point the improvement obtained in the environment is $\Delta\overline{D_f}(50\%) = 13.20\%$. From this point and depending on the size of the controllable users, $\overline{D_f}$ value grows for $n \leq 50$ and decreases in less than 0.1 % for the rest of n. Anyway, in the best case the maximum value reached by $\overline{D_f}$ is 1 and $s[k]$ is improved in $\Delta\overline{D_f}(100\%) = 13.72\%$.

In the weekly case, the starting point of the curve is higher than for the daily period. The reason is that during the week, $s[k]$ is closer to demand the full capacity of the system. The value of $\overline{D_f}$ at $L_C = 0\%$ is equal to 0.936. Then, this values grows until it reaches the elbow point of the curve for $L_C = 50\%$. At this point, the EDeNCs have improved the value of $\overline{D_f}$ in $\Delta\overline{D_f}(50\%) = 6.49\%$, which is lower than the previous evaluation period since $\overline{D_f}$ starts closer to the maximum value. As in the daily evaluation period, the best value of $\overline{D_f}$ is obtained for $n \leq 50$ and for the rest of the sizes, the values differ in less than 0.1 %. Thus, for the maximum L_C, the reached value of $\overline{D_f}$ is equal to 1 and means a final improvement of $\Delta\overline{D_f}(100\%) = 6.82\%$.

L_C[%]	Day			Week			Month			Year		
	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$
0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	1.86	2.14	0.15	1.91	2.24	0.00	1.89	2.24	0.00	1.82	2.19	0.00
20	5.00	5.78	1.53	6.32	7.41	0.02	6.32	7.42	0.00	6.17	7.30	0.00
30	7.48	8.68	8.67	11.53	13.82	3.80	12.52	15.12	2.62	14.63	18.09	0.00
35	**8.07**	**9.40**	10.40	**12.71**	**15.38**	4.72	**13.92**	**17.00**	3.26	**16.52**	**20.78**	0.00
40	8.44	9.85	11.75	13.47	16.39	5.51	14.85	18.29	3.81	17.86	22.77	0.00
50	8.88	10.38	**13.20**	14.20	17.40	**6.49**	15.78	19.60	**4.52**	19.31	24.99	**0.00**
60	9.21	10.79	13.53	14.59	17.93	6.71	16.20	20.21	4.68	19.84	25.82	0.00
70	9.50	11.14	13.64	14.89	18.35	6.78	16.51	20.66	4.73	20.18	26.35	0.00
80	9.75	11.45	13.49	15.14	18.70	6.63	16.75	21.01	4.59	20.40	26.69	0.00
90	9.97	11.73	13.48	15.36	19.01	6.61	16.97	21.32	4.57	20.61	27.00	0.00
100	**10.16**	**11.97**	**13.72**	**15.55**	**19.28**	**6.82**	**17.17**	**21.62**	**4.76**	**20.82**	**27.33**	**0.00**

Table 5.2: Comparison of $\overline{C_f}$, $\overline{L_f}$ and $\overline{D_f}$ for all the periods of time and all controllable capacities in environment B. In bold, the results discussed throughout the text are highlighted.

The $\overline{D_f}$ for the monthly evaluation period at $L_C = 0\%$ has a value of 0.954. Like the previous periods, $\overline{D_f}$ presents the same behavior for the different sizes of controllable users. It reaches a maximum value of $\overline{D_f} = 1$ for $L_C = 100\%$, which means a maximum improvement of $\Delta\overline{D_f}(100\%) = 4.52\%$. Whereas at the point of tendency change $L_C = 50\%$, the algorithm achieves $\Delta\overline{D_f}(50\%) = 4.76\%$ quite near to the maximum improvement. As for the previous coefficients, the results are very similar to the two environments, although the tendencies in the graphical representations differ. For all the periods and L_C, the results are collected in Table 5.2.

During the evaluation assessment of environment B, little differences have been found with respect to the results of environment A. Despite the difference in scale, τLA is able to found an optimum working point in all situations. Thus, the ensemble of evolved neural controllers is coordinated in order to flatten the response of $s[k]$. However, some differences are appreciated as the tendencies and the different $\overline{C_f}$, $\overline{L_f}$ and $\overline{D_f}$ curves present different forms. In addition, the last result of the assessment is the flatness of the environment through its load duration curve. Figure 5.15 represents the load duration curve for different percentage of controllable users. The curves have been normalized in order to compare them. In this case, τLA has reached different solutions than the ones obtained in Figure 5.11 for environment A. This confirms that τLA is able to coordinate the ensemble of individuals searching for the best solution to flatten $s[k]$.

As stated before, the inclusion of any percentage of controllable demand improves the behavior of the system and reduces the variability of the grid. The slope of the curves in Figure 5.15 is reduced as L_C grows and gets flatter with the increasing controllable demand. It can be also observed how the maximum consumption of the system is reduced since the system efficiency increases with flatness of the grid environment. Regardless the size of the environment, the coordination of the different users always improves its status. As in environment A, it can be observed the fast improvements introduced by relatively low values of the controllable fraction ($L_C \leq 20\%$). And the analysis of the higher demand for the first 10 h leads to an important reduction of the system marginal costs in order to meet the maximum power demand. To put some numbers to Figure 5.15, in the case for $L_C = 30\%$, the slope of the curve has been reduced in a 81.60 % with respect to the case of $L_C = 0\%$ and the maximum consumption of the environment has been reduced in a 28.93 %. For the maximum controllable capacity ($L_C = 100\%$), the curve is constant, the reduction of the slope

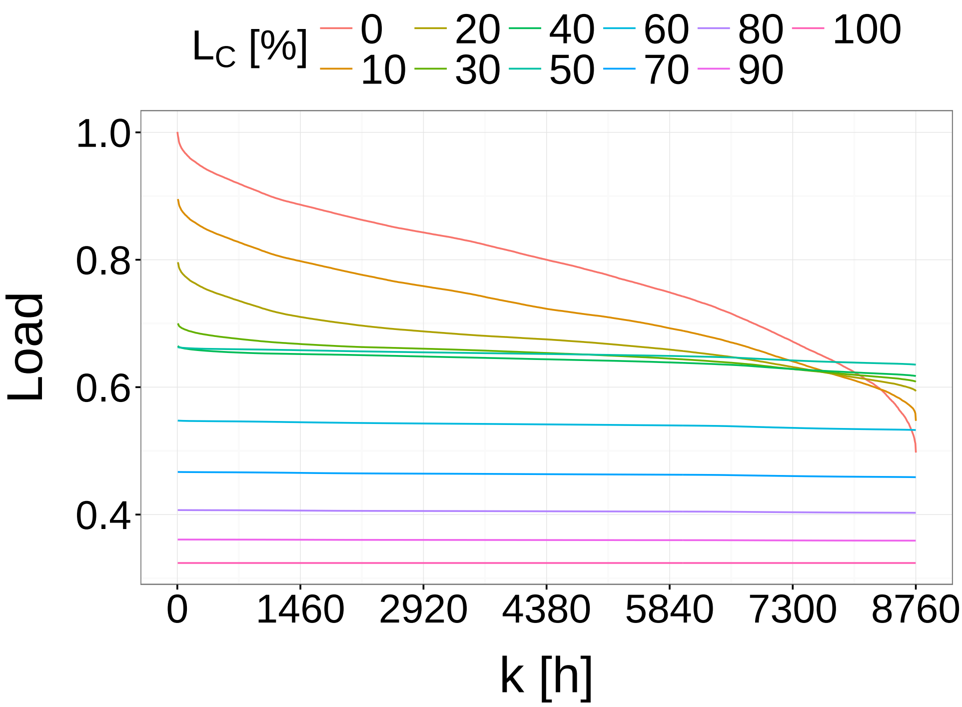


Figure 5.15: Load duration curve of environment B for different controllable load capacities.

is 100 % and the maximum reduction is 66.29 %. In this case the results are different from the ones obtained in environment A. Comparing Figures 5.11 and 5.15, the load duration curves differ from the different environments. In this case, it has been obtained an increasing reduction for all the controllable percentages in the slope of the curves and the maximum achieved.

5.4 Summary and discussion

In this Chapter some progress has been made towards the solution to the problem considered in this Thesis. The complexity level of the grid environment has been increased by adding more users to it. Specifically, there are m users for the non-controllable part of the demand and n controllable users. Both types of facilities only consume from the grid since firstly the problem of the collectivity has to be faced. The problem is focused to solve the interaction of a heterogeneous environment comprised by different quantities of users in order to reduce the variability of the aggregated consumption.

A small part of the problem has been already solved in Chapter 4 with a simplification of the environment consisting of only two users, each one for each type of demand. Thus, the next natural step to solve the whole problem was to use the solution reached in this new environment comprised by more users. However, the results of applying directly the same evolved neural controller for all the controllable users were not completely satisfactory. The reason was that all of them were trying to consume the same power at the same time, so more peaks appeared in the waveform of the signal. It was necessary to develop a mechanism that will adjust the response of the controllable users in real time, to modify their demand profile. In addition, this mechanism had to respect the essence of the evolved neural controllers in order to obtain a destructive interference through cancelling the derivative of $s[k]$.

Based on these premises and inspired by the Blind Source Separation (BSS) techniques, the development of the evolved neural controllers arose in the form of an algorithm that adjusted the value of the last free parameter of the Continuous Time RNN (CTRNN). This last free parameter is the τ_i, which is related with the speed reaction of the neurons. Thus, the algorithm displaces the output of the evolved neural controllers through changes in values of τ_i. It is necessary to assure that the stability of the system has not been comprised by the changes of τ_i. So, the algorithm has a range of values in which the stability is guaranteed. This algorithm was inspired by a reinforcement learning scheme, in which the analysis of the environment status $s[k]$ gives a measure of its flatness to alter the value of τ_i. The measurement to modify the τ_i was based on the difference of $s[k]$ to its mean value. Thus, the algorithm tries to minimize this difference through a descent gradient technique and the result of the minimization is used as the quantity to modify τ_i. The rate at which each τ_i learns for each controller is randomized by following an exponential distribution.

With the use of τ-Learning Algorithm (τLA), the development of the evolved neural controllers was achieved in this new environment. However, it was necessary to test that τLA was the proper solution to the problem. Thus, an evaluation process was defined in which three factors were analysed: i) crest factor (C_f), ii) load factor (L_f) and iii) demand factor (D_f). The first one is a direct measure of the waveform, normally $C_f \geq 1$. The other two are environment utilization factors that measures its efficiency with respect to the maximum capacity of the system, they take a value $L_f \leq 1$, $D_f \leq 1$. Moreover, two modes of actuation were defined for the test: environment A, in which the maximum power was constrained, and environment B, in which there was no maximum power limitation. For the first environment, the controllable users had to adapt their output to the behavior of the non-controllable users in order not to surpass the maximum power available. The best chromosome obtained in Chapter 4 was used for the Evo-Devo Neural Controller (EDeNC). On the other hand, there is not any power limit in environment B, so the neural controllers can consume what they need to flatten $s[k]$. For this environment, it is necessary to normalize the CTRNN input so the neural controller could respond to the environment. But, this required to re-evolve the neural controller since the boundary conditions have changed.

The results obtained during the evaluation process proved that the use of the τLA for the EDeNCs improves the status of the environment for any L_C. In addition, no matter the seed used, the τLA converges to a similar solution in all the cases. Even for different sizes of controllable users at a fixed L_C, the algorithm was able to find the optimum solution for all of them. For environment A, from $L_C = 40\%$ the environment signal smoothed its waveform in a 80 % while in environment B this cipher was reached from $L_C = 30\%$. The maximum flatness of the environment signal was achieved for $L_C = 100\%$ and it was of a 100 %. Four periods of evaluation were defined for the 3 factors in both environments for all the possible configurations. The best value of each parameter was achieved in both environments for the highest controllable capacity $L_C = 100\%$, i.e. $C_f \approx L_f \approx D_f \approx 1$. The improvement of each coefficient for the different L_C was gathered in Tables 5.1 and 5.2 for environment A and B, respectively. In conclusion, the EDeNCs using τLA were able to achieve a constant $s[k]$ which improves the performance of the environment, working always at the full capacity of the system.

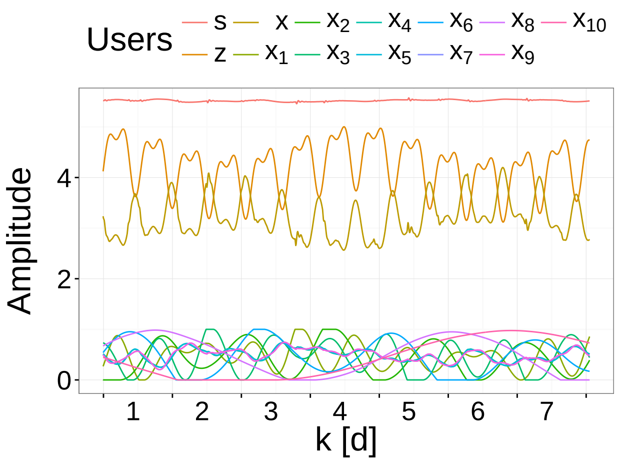
6 Neural Grid

"I'm gonna show you, how great I am" — *Muhammad Ali*

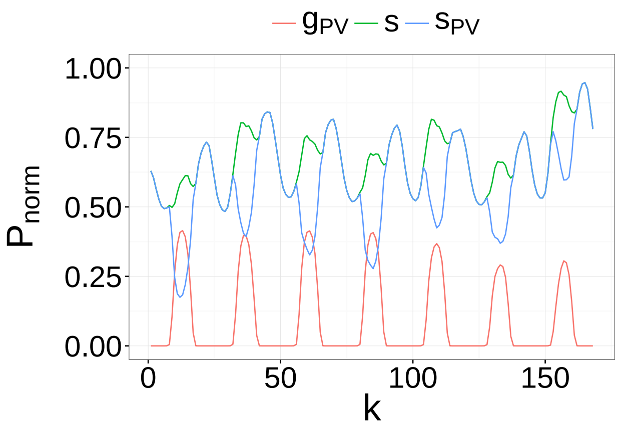
Regarding to the results obtained in Chapter 5, the ensemble of neural controllers was able to smooth a synthetic demand curve based on the principal frequency components of a real aggregated consumption. The neural controllers were evolved to obtain the opposed derivative of the input signal (aggregated consumption) and cancel the variability of the continuous signal at their input. After that, they were developed using the τ-Learning Algorithm (τLA) in order to cancel collectively the environment signal (aggregated consumption). A heterogeneous response was obtained for the different controllers when the algorithm was applied. Figure 6.1(a), shows an example of the behavior of 10 users in which the controllable demand is divided. Each of them presents a different demand profile and a flatten aggregated consumption is achieved when added to the non-controllable demand. Although this environment is close to a real electric grid, it is still far from its implementation. In this Chapter, the level of complexity is increased to test the development of the evolved neural controllers in an environment with real demand and Distributed Energy Resources (DER). To this extent, the possibilities of using this kind of coordination algorithms are analyzed in real environments. In addition, the inclusion of DER elements gives the opportunity to assess the interaction of local generation and demand, and how the controllable capacity would help to the DER integration inside the grid.

So far, neural controllers were able to use all the power available to cancel the variability of the non-controllable demand. However, the necessities of the users, controllable and non-controllable, must be met. Thus, the neural controllers are restricted to use only the power from the deferrable loads which can be displaced around the time axis. Moreover, a new criterion to displace the power of deferrable loads is developed by adding local energy resources capabilities. The controller has to take into account this new dimension of the problem in order to smooth the aggregated consumption plus the maximization of the local energy resource. These two objectives are in conflict a priori. The reason is that the maximum production of Photovoltaics (PV) electricity is located at the central hours of the daylight concurring with one of the daily maximum consumption periods. Thus, if the consumption of all the users were supplied locally by PV generators, the aggregated consumption would present a form such as the one of Figure 6.1(b). In this Figure, it is represented the generated power $g_{PV}[k]$ and the aggregated consumption with or without taking into account the effect of the PV generation, $s_{PV}[k]$ and $s[k]$. The waveform of $s_{PV}[k]$ in Figure 6.1(b) presents a higher variability than the current behavior of the grid, $s[k]$. Thus, the algorithm needs to confront a higher variability and discontinuity of the environment signal, which makes difficult the smoothing of the aggregated consumption.

In this Chapter, a new approximation for the neural controllers is described in order to meet these new restrictions. An improved algorithm is built based on the use of the neural controllers described in Chapters 4 and 5 and the addition of the users behavior plus the DER elements. This algorithm uses the demand profiles generated by the Evo-Devo Neural Controllers (EDeNCs) to smooth the aggregated consumption in a distributed way. The EDeNC consumption patterns are used to schedule the deferrable loads that each user allows controlling. Moreover, the



(a)



(b)

Figure 6.1: Grid behavior for different environments: (a) example of the τ-learning algorithm to divide the antiphase controllable demand into 10 different users and (b) effect of the PV generation in the aggregated consumption. In red the PV generation, g_{PV}, in green the environment signal without PV, s, and in blue the environment signal with PV, s_{PV}.

algorithm combines the EDeNC patterns with the local generation for the scheduling, since all consumers have PV generators. The algorithm builds a pattern based on the response of the EDeNC and the PV generation forecast. Thus, the algorithm seeks the balance between the smoothing of the aggregated consumption and the maximization of the self-consumption, which focuses on the maximum use of the own generated energy, while the energy provided by the grid remains an optional generator or consumer.

The GridSim[1] simulator has been used to develop this new algorithm and to carry out the final tests of this new algorithmic approximation. It is an open source simulator in which the power balances of a virtual electrical grid are analyzed. GridSim simulator was introduced in Castillo-Cagigal (2014). Hence, this simulator incorporates all the elements necessary to simulate the real behavior of the users and

[1]Source: https://github.com/Robolabo/gridSim

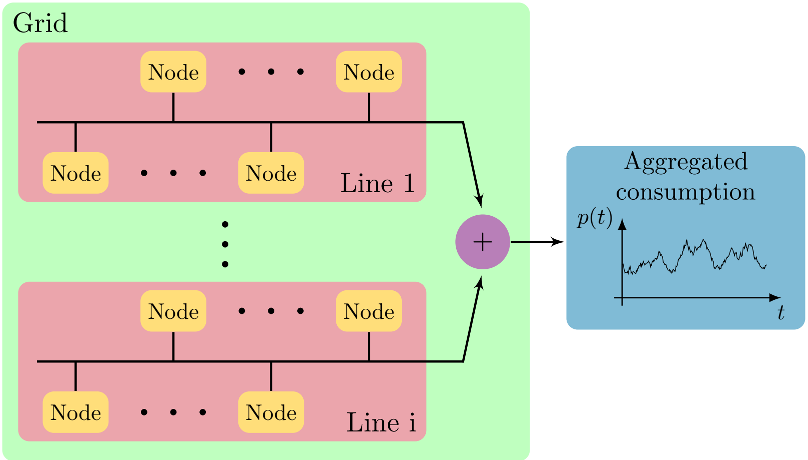


Figure 6.2: Schema of the main elements of the simulator GridSim.

the different elements of the grid into a virtual environment. A brief explanation of the simulator and its elements is included in Section 6.1. Then, the Neural Grid Algorithm (NGA) which is the new algorithm developed using the EDeNC response is presented in Section 6.2. The scheduling of loads based on different profiles is explained together with the control strategy followed. After that, simulations are held to test the NGA approximation and the final tests of the EDeNC for its application in a virtual environment (see Section 6.3). Finally, a conclusion of the Chapter is carried out in Section 6.4.

6.1 GridSim: virtual grid environment

GridSim is an open source software which simulates the real behavior of a grid referred to the power balance of its components. With this piece of software, simulations of real environments are carried out to test the NGA. The structure of the simulator consists of a group of nodes which are connected together through a virtual power line. And the set of virtual power lines are connected together creating the structure of a virtual grid (see Figure 6.2). Each node is composed by different elements:

- A *consumption profile*, whose form depends on the energy necessities of the user. Different profiles can be used in order to satisfy different users necessities. So, the profile depends on the user selected and the loads that it possesses.
- A *DER system* consisting of a local generator with or without an storage system. In this Thesis, a PV system is used as source of the Distributed Generation (DG) and the storage system is composed by a lead-acid battery. Despite not being the most efficient technology, lead-acid batteries are the most known, widely used and economic technology. GridSim also offers the possibility of incorporating other DER technologies, by changing the models used and the different parameters of the element.
- A *controller* can also be added to manage the different power flows that are concurring inside the node by establishing different objectives to modify the node behavior. Different controllers can be used, for example for battery management or for Demand Side Management (DSM) of a set of local loads.

Thus, different types of nodes can be defined by combining the different elements inside them. For example, there can be nodes with only consumption or generation, nodes with both of them, nodes with controllers to manage the power flows inside the virtual grid, back-up nodes with storage systems to store electricity or any possible combination among them. To simplify the operations inside the grid, each line groups together only one type of node. So, the nodes inside of a line have the same configuration and possess the same elements. Then, *GridSim* calculates the power balances of every node and they are aggregated in their common line. And the aggregated consumption is calculated as the sum of the power balances of each line.

The configuration of the nodes varies depending on the situation simulated. The simulations of this Chapter consist of analyzing the behavior of the NGA in a closer environment to the Smart Grid (SG). To this end, the nodes are equipped with PV and local storage systems that constitute their DER. Moreover, the nodes will be also provided with controllers based on the previous development of the EDeNC. However, the size of the controllable population will depend on the controllability capacity of the grid. Hence, the population of users will be divided in controllable and non-controllable ones.

6.1.1 Consumption profile: virtual user

The consumption profile of the users is composed by the three types of loads presented in Section 2.3.4. The representation of these three loads inside the simulator is presented in Figure 6.3. There are some differences between them, the elastic loads consume instantaneously the power that they are demanding (e.g. corresponds to air-conditioning loads or Electric Vehicle (EV)). On the other hand, deferrable and non-deferrable loads are discretized in energy packages of determined power amplitude (P) and time length (Δt, see Figure 6.3). Their mathematical expression is gathered in Equation 6.1.

$$p^L_{i,j}(t) = \begin{cases} P & t \in [t^{act}_{i,j}, t^{act}_{i,j} + \Delta t] \\ 0 & t \in rest \end{cases} \tag{6.1}$$

where $p^L_{i,j}(t)$ is the instantaneous power consumed by the load (deferrable and non-deferrable) for the jth load of the ith user, $t^{act}_{i,j}$ is the activation time of the load, P is the power consumed by the load and Δt is the duration of the load. Apart from the power consumed by the load, the difference between deferrable and non-deferrable loads is the time at which the load starts ($t^{act}_{i,j}$). For non-deferrable loads, this time is fixed and cannot be displaced by the controller. For deferrable loads, this time is not set and can be assigned within the user preferences inside a period of time given by him or a *running range* ($\Delta t^u_{i,j} = [t^{beg}_{i,j}, t^{end}_{i,j}]$).

This representation of the node loads makes easier to model any type of user, residential, commercial or industrial. It is possible to simulate their behavior by knowing the amplitude of the loads, their duration and the time at which the different loads are activated. For this Thesis, the nodes are going to simulate residential users, so the energy packages are going to represent the different appliances and home electronics devices that they possess. In the example of Figure 6.3, the form of the three types of loads can be observed. The elastic loads in purple present a continuous profile whose form evolves through time (for representations purposes, the elastic consumption profile is discretized to make easier its aggregation to the other loads). Non-deferrable loads present a discrete profile consisting of different energy packages representing different loads whose $t^{act}_{i,j}$ is fixed by the user preferences. Finally, the deferrable loads are represented in green. In this case, each energy package also represents different loads but the $t^{act}_{i,j}$ can be displaced in time inside the limits by a controller. The main difference between these two types of loads in this representation is the time when the load starts and how this time is assigned. As observed in Figure 6.3, a non-deferrable load is a deferrable one whose limits of the $\Delta t^u_{i,j}$ are $t^{act}_{i,j}$ or in other words, there is no interval to select the time. The consumption profile of the node consists of the aggregation of all the loads inside it and presents the behavior of the simulated user.

GridSim also incorporates a data base of appliances profiles and other residential consumptions to simulate the concrete behavior of a residential user. These residential

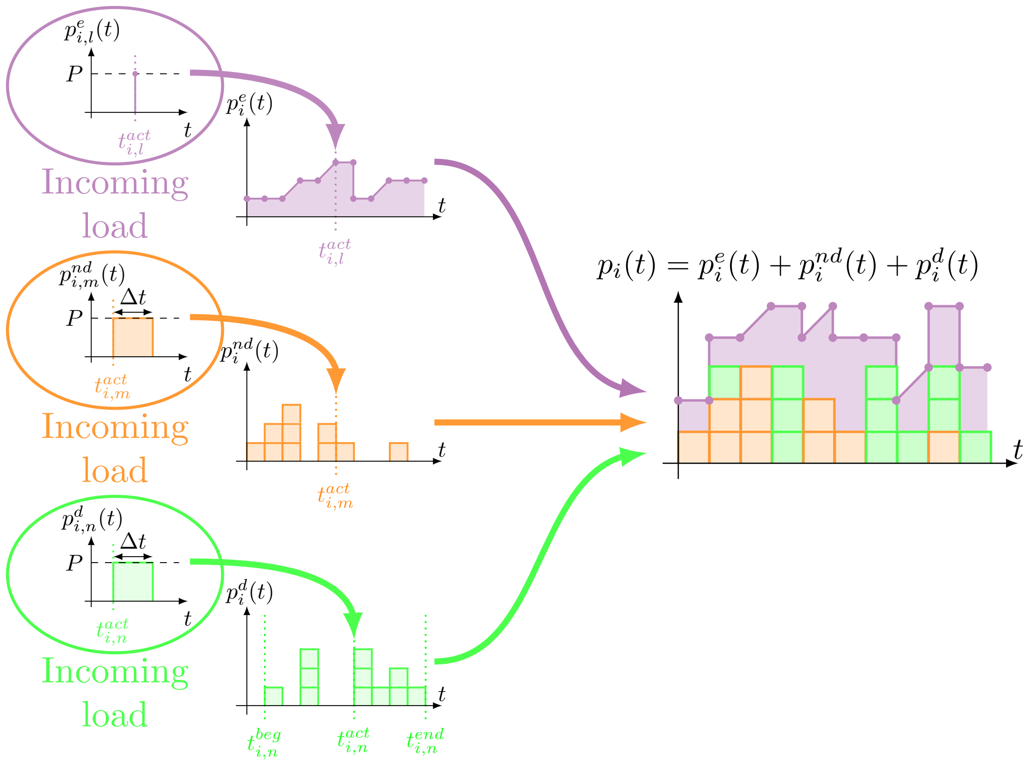


Figure 6.3: Consumption profile for a user with different load types: elastic, non-deferrable and deferrable. In purple the elastic loads, in orange the non-deferrable loads and in green the deferrable loads.

loads include the typical electrical appliances of a highly electrified house. The non-controllable loads of the data base are: cooking appliances, fridge, freezer, lighting, computers and entertainment appliances. While there are only three deferrable appliances: washing machine, dishwasher and dryer. All the information was measured in the demonstrator MagicBox with a time resolution of a minute and these data was gathered in Castillo-Cagigal et al. (2011b). For the simulations, the energy package discretization is used since any electrical behavior can be represented. Thus, how is GridSim able to place these packages in order to simulate the behavior of an entire grid? The loads of the node that can be placed in the time axis are the deferrable and non-deferrable loads. Deferrable loads have a flexible $t^{act}_{i,j}$ and they will be scheduled following the given $\Delta t^u_{i,j}$ by a controller. This process will be explained in Section 6.2. However, if there are no controllers, these deferrable loads are assigned in the same instant of their creation and will follow a consumption profile as the non-deferrable ones since the tasks have to be done.

In order to simulate the aggregated consumption of a grid, a virtual user is used. This virtual user consists of the creation of random loads whose $t^{act}_{i,j}$ is assigned by using a local consumption pattern, defined as $f_{L,i}(t)$. Thus, deferrable loads are used to follow the consumption pattern of a grid. $f_{L,i}(t)$ defines the shape of the user local consumption and through the aggregation of all users, the global behavior should be similar to the one of a real electrical grid. To this extent, the different loads are positioned in the time axis following the local pattern which is used as a probability density function. Thus, $t^{act}_{i,j}$ is considered as a random variable whose value is taken with a certain probability from $pdf(t^{act}_{i,j})$. In this case, the user has a number of loads that have to be done per day, which are used or not daily according to the user's

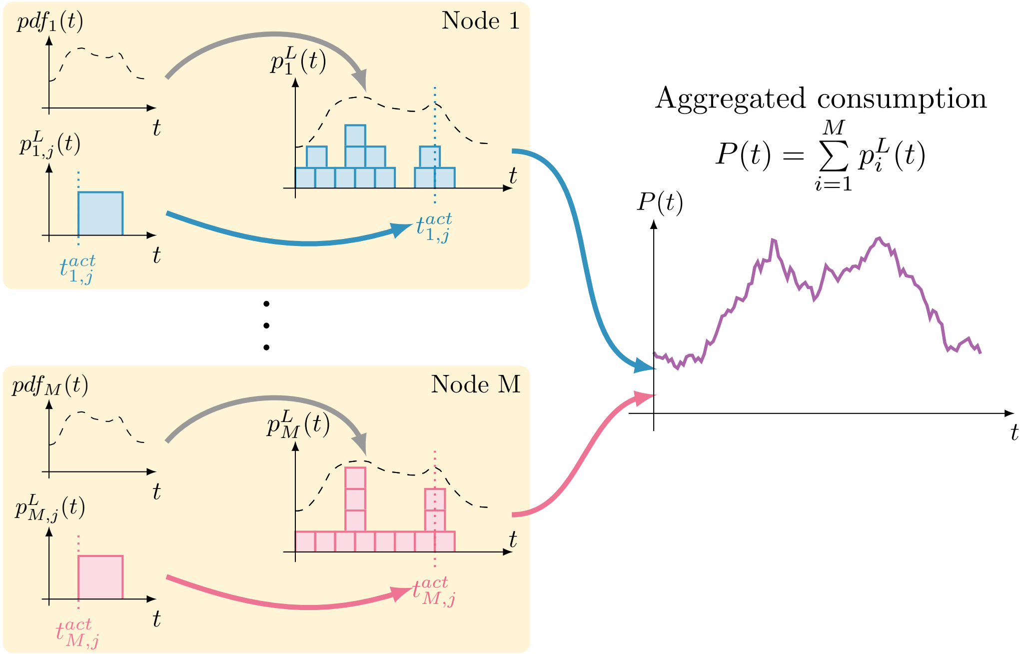


Figure 6.4: Example of positioning loads based on the $pdf(t_{i,j}^{act})$ of the desired global behavior.

needs. Then, the mathematical expression of the $pdf(t_{i,j}^{act})$ is represented in Equation 6.2.

$$\begin{aligned} pdf(t_{i,j}^{act}) &= \frac{1}{C} \cdot f_{L,i}(t) \quad \text{for} \quad t \in [t_{i,j}^{act}, t_{i,j}^{act} + t_{day}] \\ C &= \int_{t_{i,j}^{act}}^{t_{i,j}^{act}+t_{day}} f_{L,i}(t)dt \end{aligned} \tag{6.2}$$

where C is a normalization factor of $f_{L,i}(t)$ since $pdf(t_{i,j}^{act})$ is a probability density function and t_{day} is the length of $f_{L,i}(t)$ at which the loads have to be placed. In this case, the length of a day in minutes is used ($t_{day} = 1440$).

Thus, if the global behavior of the users should be similar to the aggregated consumption of a grid, the $f_{L,i}(t)$ provided to the user will be the aggregated consumption itself. The reason is that in probability the user should consume more in the peaks than in the valleys following the shape of the aggregated consumption. One user does not have enough loads to represent it, but the aggregation of all users will give the expected result of a similar aggregated consumption. Figure 6.4 shows the behavior of M virtual users and their aggregation. In this Figure, the incoming loads are activated based on the day aggregated consumption of a grid. They are distributed by following the patterns provided and new incoming loads are placed in the most probable time to occur. $pdf(t_{i,j}^{act})$ is calculated by using the activation time of the load and the next 1440 min (the length of a day in minutes) to place it in the most probable time. Two examples of load assignation are described in Figure 6.4 for nodes 1 and M. Both profiles present different shapes and through the aggregation of the M nodes, a grid aggregated consumption similar to the real one is obtained.

Finally, depending on the node, the loads can be rescheduled with a controller following the user preferences, $\Delta t_{i,j}^{u}$, and change their consumption profile. Thus, the aggregated consumption changes depending on the objective of the controller. However, if the node does not have any controller, the consumption remains the same without any change. So the virtual user will provide the local consumption of each node and the use of controllers will alter the grid behavior.

6.1.2 Distributed Energy Resources module

GridSim incorporates two modules related with DER: i) generation and ii) storage systems. The different elements (consumption, generation, storage system and grid) inside a node are connected following an Alternating Current (AC) bus topology such as the one of Figure 4.5. The generation module inside the node is used to simulate DG. This module only incorporates PV generation and can be configured with different sizes of generators. It also incorporates the elements inside a PV system, such as inverters. The generation module also takes into account the losses and efficiencies of the system through the configuration of those parameters. GridSim is able to calculate the output power of a PV system from the environment data (solar radiation and temperature), the generator parameters (size, orientation, inclination, etc.), losses factors (voltage drops, thermal, shadows, etc.) and conversion efficiency of the inverters. So, it gives the possibility to simulate any PV generator of flat silicon PV modules.

In addition, the generation module of GridSim has an input mode in which the output power of a real system can be introduced to simulate different types of generation. Thus, different sources of generation can be added to the simulations in order to emulate the generation mix of real electrical grid. It is also possible to change the PV penetration by adding more nodes. Moreover, the isolation of generation from the consumption is also possible through changes in the grid configuration by defining a line with only generation and no consumption. A line can be defined per type of generation by adding nodes with no consumption plus an input file with the power output. GridSim gives a lot of possibilities to simulate different types of generation and their configuration in order to understand the different power flows inside an electrical grid. In order to study the maximization of the local energy resources, simulations are done using the PV DG systems through the use of real grid measurements. Specifically, these data correspond to the PV generation measurements of the Spanish peninsular system during the year 2015.

The other DER module available in GridSim that completes the node configuration is the storage system. The simulator allows configuring an storage system consisting of a battery and a bidirectional inverter that allows charging and discharging the battery. In order to resemble to the real functioning of an storage system, GridSim allows configuring the conversion efficiencies of the inverter, the size of the storage system in days of autonomy, the state of charge of the battery, the nominal voltage, etc. At the moment, only lead-acid batteries are modeled and programmed inside the simulator since their behavior are well-known and this technology is mature enough.

The storage system in GridSim is configured by default as a backup system which is only discharged when the node is isolated from the grid and there is no power source to feed the loads. In addition, the battery is only able to charge from local energy sources and never from the grid. In this way, GridSim avoids some questionable transactions with the system, such as buying cheap electricity to sell it when it is more expensive. By default, the battery voltage is $48\,V$ and the capacity is 1 day of autonomy and the minimum State of Charge (SoC) is $40\,\%$. These parameters have been selected based on the studies of Castillo-Cagigal et al. (2011a).

All the storage system parameters can also be modified from the corresponding configuration file and other batteries with new materials can also be added. Moreover, the behavior of the storage system in the node can be altered by adding a controller in charge of managing the battery. Thus, the storage can be used not only as a backup system, but also as a local resource to supply the loads when the power of the local generator is not enough. Hence, in this case the grid is used as the backup system, only supplying power to the loads when there is not enough power supplied locally. In addition, the grid also receives the surplus of electricity generated that cannot be used to supply the loads or to store in the battery. Thus, the exchanges with the grid are reduced and the autonomy of the node is increased.

As the generation module, the grid topology could incorporate only storage systems that serve as backup to store the surplus of generated electricity. Therefore, the inclusion of distributed storage capacity should serve to relax the condition to meet the demand since they can be used to adapt the electricity production to the demand. Moreover, distributed storage increases the self-consumption of local energy resources, displacing the extra generated electricity to those times where the electricity is needed. In this Chapter, the influence of these DER elements are studied from the perspective of the grid global behavior. To that extent, different simulations environments are configured.

6.2 Neural grid: scheduling with ANN and PV generation

The node configuration allows the addition of a controller to manage the different power flows circulating inside it. Based on the EDeNC approximation explained in Chapters 4 and 5, a new algorithm is designed to manage the node power flows and it is called Neural Grid Algorithm (NGA). The NGA is going to reschedule the deferrable loads of the local facility in order to achieve multiple objectives: i) smoothing the aggregated consumption and ii) maximizing the use of local energy resources or maximizing the self-consumption. Thus, NGA looks for reducing the global variability of the grid at the same time that maximizes the self-consumption of the local energy sources. The facility is composed by the GridSim elements already presented, a discretized consumption in energy packages and the DER.

In order to schedule the different loads, NGA will assign $t_{i,j}^{act}$ of each deferrable load within $\Delta t_{i,j}^{u}$ selected by the user. So, when a new deferrable load is created, NGA is fed with the user preferences and it will define $t_{i,j}^{act}$ based on the time preferences, assuring that time restrictions are satisfied. The algorithm uses a local consumption pattern in order to assign $t_{i,j}^{act}$ ($f_{L,i}(t)$, see Section 6.1.1). At the same time, $f_{L,i}(t)$ can be composed by different shapes which identify different objectives to schedule the loads. Hence, NGA uses $f_{L,i}(t)$ as the objective form that the local consumption should resemble and it is also used as $pdf(t_{i,j}^{act})$ to schedule the loads. As in Section 6.1.1, $t_{i,j}^{act}$ is considered as a random variable which takes a value inside $\Delta t_{i,j}^{u}$ with a certain probability given by $pdf(t_{i,j}^{act})$. In this case, the $pdf(t_{i,j}^{act})$ form introducing $\Delta t_{i,j}^{u}$ in its definition and the temporal limits of Equation 6.2 are changed. This new form is gathered in Equation 6.3.

$$
\begin{aligned}
pdf(t_{i,j}^{act}) &= \frac{1}{C} f_{L,i}(t_{i,j}^{act}) \quad \text{for} \quad t_{i,j}^{act} \in [t_{i,j}^{beg}, t_{i,j}^{end}] \\
C &= \int_{t_{i,j}^{beg}}^{t_{i,j}^{end}} f_{L,i}(t)dt
\end{aligned}
\tag{6.3}
$$

where C is the normalization constant which guarantees that $pdf(t_{i,j}^{act})$ is a probability density function. Thus, NGA will be placing the deferrable loads following the desired local consumption pattern.

NGA provides the deferrable loads with the local pattern based on the two objectives previously stated: flattening the aggregated consumption and maximizing the self-consumption. Thus, the $f_{L,i}(t)$ is also divided in two parts. The first one corresponds to the first objective and its shape is the output of the EDeNCs, $f_{L,i}^{EDeNC}(t)$. The EDeNCs have already proven that can cancel collectively the variability of a signal. While the second one is associated with the local generation and it will have the form of the PV generation forecast, $f_{L,i}^{PV}(t)$. Therefore, $f_{L,i}(t)$ has the expression

$$
f_{L,i}(t,\beta) = \beta \cdot f_{L,i}^{PV}(t) + (1-\beta) \cdot f_{L,i}^{EDeNC}(t) \tag{6.4}
$$

where $\beta \in [0,1]$ is the parameter used to adjust the priority or the importance of the objectives in the local consumption pattern. This means that for $\beta = 0$ the algorithm prioritizes the smoothing of the aggregated consumption since the local pattern tends to be equal to $f_{L,i}^{EDeNC}(t)$. While for $\beta = 1$, the algorithm is prioritizing the self-consumption of the local generated electricity, since the local pattern is equal to $f_{L,i}^{PV}(t)$. Notice, that for both functions, $f_{L,i}^{EDeNC}(t)$ and $f_{L,i}^{PV}(t)$, the algorithm needs their future shapes that is the reason of using the forecast PV generation. But the response of EDeNCs cannot be used directly since the controllers work on real time and some changes are required to obtain the $f_{L,i}^{EDeNC}(t)$.

6.2.1 EDeNC as consumption profile

As presented in Chapter 5, the EDeNCs generate a distributed antiphase signal based on the creation of a destructive interference that cancels the variability of a global input. To do that, the EDeNC is able to forecast one step ahead the behavior of the input, which is a signal of class C^1. It extracts the necessary information of the signal through the tendency of its derivative. However, the forecasting horizon of one step is not enough for scheduling the deferrable loads. So, it is necessary to extent this forecasting horizon. There are two possible options, the first one consists of evolving and developing the neural controllers by taking into account this new restriction, or the second one based on the same EDeNCs and adapting its output dynamically to the requirements of the process.

As explained in Section 2.1.5 and 4.1, the aggregated consumption of an electrical grid is a periodic signal in which the user behavior has a high repeatability. For example, the consumption of the different working days (Monday to Friday) has a similar shape and does not differ too much in energy terms. However, weekends present a different shape compared to the working days. Hence, analyzing the different periods of the signal through the Fourier Transformation, the result is that there is a high weekly component. For this reason, the $f_{L,i}^{EDeNC}(t)$ could be built by using the consumption profile of the previous week applied to the current one since the differences in shape and energy are low. Thus, the actual EDeNCs could still be used and its output takes into account the real time information plus the information about what happened a week earlier. Then, the EDeNCs output would be enough to place the deferrable loads in the time axis following their response. Finally, the load pattern corresponding to the controllers will be governed by Equation 6.5.

$$f_{L,i}^{EDeNC}(t) = x_i(t, \Delta t_{week}) \tag{6.5}$$

where, $x_i(t, \Delta t_{week})$ is the output of the ith EDeNC and Δt_{week} is the historical time corresponding to the previous week from which the algorithm will place the deferrable loads.

The EDeNC output is in the range $[0,1]$ which is necessary for $pdf(t_{i,j}^{act})$. And the aggregation of all facilities with this modification of the EDeNC will continue having the same properties described in Chapter 5. In addition, it will allow to continue applying the developed algorithm, τLA, since the neural controller response will adapt to the new environment and no extra modifications have to be done in the structure. Thus, the controllable users will generate an antiphase consumption pattern collectively that will oppose to the aggregated consumption of the non-controllable ones, flattening it. Therefore, when $f_{L,i}(t)$ has a $\beta = 0$, the consumption pattern of the local facility is directly the EDeNCs output used as $pdf(t_{i,j}^{act})$ and it presents the following form,

$$\begin{aligned} pdf(t_{i,j}^{act}) &= \frac{1}{C} f_i^{EDeNC}(t_{i,j}^{act}) \quad \text{for} \quad t_{i,j}^{act} \in [t_{i,j}^{beg}, t_{i,j}^{end}] \\ C &= \int_{t_{i,j}^{beg}}^{t_{i,j}^{end}} f_i^{EDeNC}(t)dt \end{aligned} \tag{6.6}$$

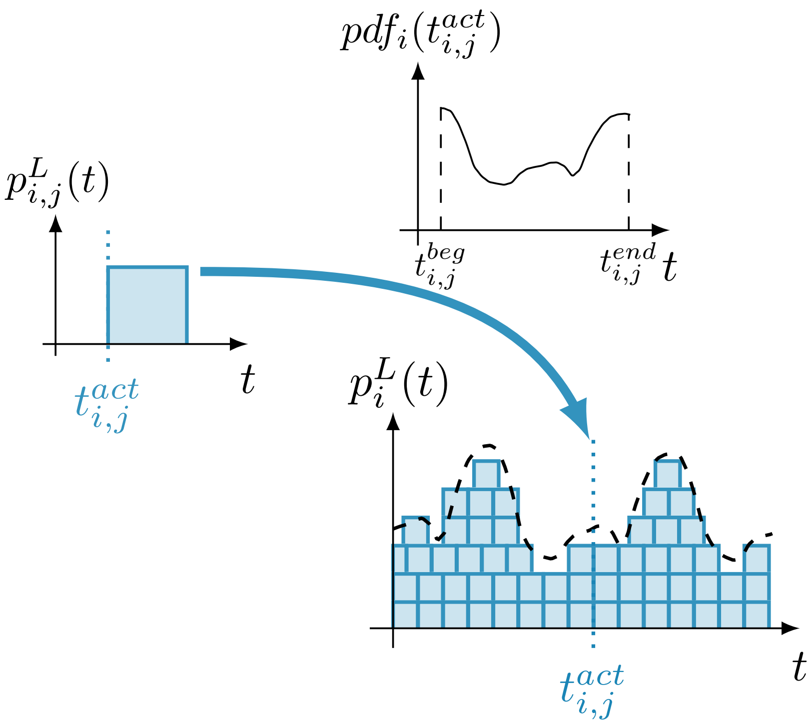


Figure 6.5: Assignment of $t_{i,j}^{act}$ for the different deferrable loads using the output of the EDeNC as local consumption pattern.

A graphical example of how the algorithm schedules loads with the EDeNC can be observed in Figure 6.5. When an incoming deferrable load arrives, it adjusts $t_{i,j}^{act}$ within $\Delta t_{i,j}^{u}$ for the probability density function obtained from the EDeNC. Thus, the local consumption will possess a discretized form in energy packages that will resemble the shape of the EDeNC output. Figure 6.5 shows how the algorithm places the different deferrable loads and its similarity to the $pdf(t_{i,j}^{act})$ used. Through the aggregation of all controllable loads, the load scheduling has the same smoothing property as the set of EDeNCs.

6.2.2 PV forecasting as consumption pattern

The second objective of the NGA consists of maximizing the self-consumption of the local energy resources. To that extent, it is necessary to include the response of the PV system inside the algorithm. However, NGA cannot use the real time production since it happens instantaneously and it needs a profile based on the future generation values in order to schedule the incoming deferrable loads. Thus, the use of PV forecast generation is required to place the load in the time axis. The more accurate the PV forecast is, the bigger the self-consumption value is since the loads will be placed inside the time of electricity production.

In order to understand how the incoming loads are placed following the PV forecast, imagine that NGA only schedules them following that profile. In this case, $\beta = 1$ and the local consumption pattern for the *ith* facility will be equal to $f_{L,i}(t, \beta = 1) = f_{L,i}^{PV}(t)$. As the consumption pattern is used to build $pdf(t_{i,j}^{act})$, it is necessary to normalize the PV generation forecast. The normalization is done by dividing the predicted values of the PV generation by the maximum generation of the PV generator. Therefore, the profile is limited between $[0, 1]$ and it is used directly by the algorithm as $pdf(t_{i,j}^{act})$ which presents the form of Equation 6.7.

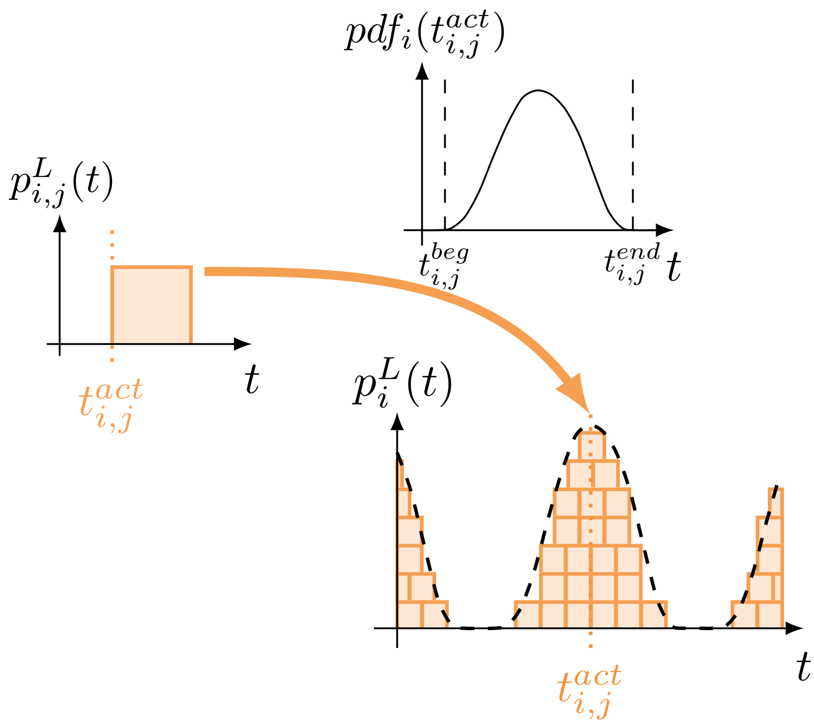


Figure 6.6: Assignment of $t_{i,j}^{act}$ for the different deferrable loads using as local consumption pattern the PV forecast of the local generator.

$$
\begin{aligned}
pdf(t_{i,j}^{act}) &= \frac{1}{C} f_{L,i}^{PV}(t_{i,j}^{act}) \quad \text{for} \quad t_{i,j}^{act} \in [t_{i,j}^{beg}, t_{i,j}^{end}] \\
C &= \int_{t_{i,j}^{beg}}^{t_{i,j}^{end}} f_{L,i}^{PV}(t)dt
\end{aligned}
\tag{6.7}
$$

In Figure 6.6, it can be observed an example of how the algorithm places the incoming loads using only the PV forecast profile when $\beta = 1$. The algorithm schedules the loads inside the generation profile given, following its bell shaped form. Following this distribution, the self-consumption of the local energy source will increase its value since the consumption matches local generation. However, the increase of the self-consumption favors the reduction of interactions with the grid which is translated in a higher variability on the aggregated consumption. The reason is that there is less consumption to aggregate during the generation period in contrast to the non generation periods in which users will continue consuming from the grid. Thus, a high difference between the two periods is created and its value depends on the PV penetration of the grid.

The elaboration of the PV generation forecast is a complex process due to the influence of different phenomena like atmospheric that alter the variability of the resource. In order to place the deferrable loads in the time axis, it is necessary an accurate forecast that takes into account the variability in the prediction, such as seasonal, daily or intradaily variabilities. So, the PV generation forecast used in the simulations consists of the one used by the Spanish grid operator for the year 2015 of the Spanish peninsular system. These forecasts take into account the different aspects before mentioned and are elaborated by grid operators for a better integration of the PV generation inside the system.

Algorithm 2 High-level description of the Neural Grid Algorithm for the facility *ith* of the grid.

1: /* Deferrable load information from the user */
2: $[\Delta t_{i,j}, \Delta t^{u}_{i,j}] \leftarrow$ Get the duration of the deferrable load and the user preferences
3: /* Calculate local consumption pattern */
4: $f^{EDeNC}_{L,i}(t) \leftarrow$ Get antiphase consumption pattern from EDeNC
5: $f^{PV}_{L,i}(t) \leftarrow$ Get normalized local generation forecast
6: $f_{L,i}(t) = \beta \cdot f^{EDeNC}_{L,i}(t) + (1-\beta) \cdot f^{PV}_{L,i}(t)$
7: /* Calculate $t^{act}_{i,j}$ */
8: $pdf(t^{act}_{i,j}) \leftarrow$ Calculate the probability density function
9: $t^{act}_{i,j} \leftarrow$ Get value of random variable with $pdf(t^{act}_{i,j})$

6.2.3 Neural Grid Algorithm

Each NGA part has been explained independently in order to understand how to schedule the loads following each one of the two objectives presented. This algorithm is conceived to help the users to decide where the deferrable consumption should be placed in the time axis following: a local objective of enhancing its self-consumption or a global objective of enhancing the status of the grid. Thus, NGA is executed in real-time and it will only be activated when a new deferrable load is going to be added by the user.

In order to be executed, the algorithm requires some information: i) the deferrable load duration and ii) the time preferences of the user in which the load will be executed. Therefore, it is the user who will fed the system with this information. Once the information is supplied, the algorithm will calculate the load consumption profile for the temporal limits given by $\Delta t^{u}_{i,j}$. First, the response of the EDeNC is obtained in the form of the $f^{EDeNC}_{L,i}(t)$. In spite of executing NGA asynchronously, the EDeNCs are executed synchronously to obtain the antiphase profile, used after to schedule the incoming deferrable loads. Thus, NGA will obtain from the response of the EDeNC, a one week ahead prediction of the antiphase aggregated consumption. After that, the local generation resources optimization is calculated. The algorithm will obtain $f^{PV}_{L,i}(t)$ from the PV forecast, calculated as in Equation 6.7. Then, both responses will be used in order to calculate the local consumption profile $f_{L,i}(t, \beta)$ as in Equation 6.4. And $pdf(t^{act}_{i,j})$ to schedule the loads is obtained for $\Delta t^{u}_{i,j}$ preferences following Equation 6.3. Finally, $t^{act}_{i,j}$ of the deferrable load will be assigned from $pdf(t^{act}_{i,j})$ as the execution of a random variable.

A descriptive summary of the algorithm is gathered in Algorithm 2. All the steps to assign $t^{act}_{i,j}$ of the incoming deferrable load are gathered in this Algorithm. In addition, an execution example of NGA is described graphically in Figure 6.7. In this example, the local consumption profile used by the algorithm is a combination of the two functions that compose the $f_{L,i}(t)$. The β used in this case has a value of 0.5 and its $pdf(t^{act}_{i,j})$ is a combination of $pdf(t^{act}_{i,j})$ used in Figures 6.5 and 6.6. Some differences are observed with respect to $pdf(t^{act}_{i,j})$ of previous examples. In this case, a local maximum appears at the valley of the antiphase aggregated consumption which corresponds to the maximum of the PV forecast generation. And finally, NGA places the incoming deferrable loads following this consumption profile and the result is a discretized consumption pattern which follows $pdf(t^{act}_{i,j})$.

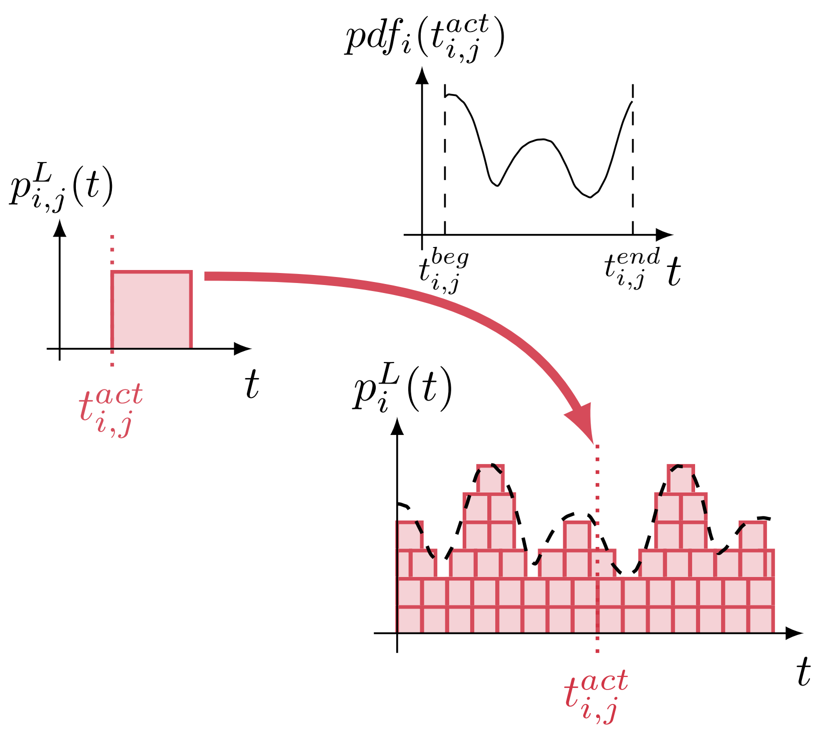


Figure 6.7: Assignment of $t_{i,j}^{act}$ based on the combined consumption profiles selected by the algorithm.

6.3 Simulation Results

After enunciating the proposed NGA, it is necessary to test the hypothesis suggested. The algorithm is analyzed in a simulation environment carried out in GridSim (see Section 6.1). The evaluation process consists of a series of simulated experiments in which the benefits of using NGA in an electrical grid are tested. To that extent, each experiment analyzes different parts of the algorithm in different environments where it could improve the grid status. The simulations were divided into three different phases:

- *Analysis of the EDeNC response.* In this set of simulations, the part of the NGA corresponding to the flattening of the aggregated consumption is analyzed. Thus, each facility will be composed only by loads and there will be no local energy resource available to feed them. In order to test the response of the NGA, the controllable fraction L_C of the facilities will be varied. However, the EDeNC response has not been tested in a real environment. Thus, a first experiment will consist of using directly the EDeNC over elastic loads and varying the L_C in order to smooth the aggregated consumption (see Section 6.3.3). Once the controllers have been tested, in the next experiment NGA will schedule the deferrable loads only by using the delayed consumption profiles given by the EDeNC, so $\beta = 0$ (see Section 6.3.4).

- *Analysis of the NGA with DG.* In this case, PV generation is added locally to all facilities without an storage system. This set of experiments will evaluate the NGA capacity to meet the two proposed objectives, the smoothing of the aggregated consumption and the maximization of the local energy sources. To this end, L_C and PV_p will vary and NGA will search for the equilibrium between both objectives by scheduling the deferrable loads properly. In a first experiment, NGA will integrate PV generation by only scheduling loads by following the smoothing of aggregated consumption (see Section 6.3.5). After analyzing the variability that PV introduces, NGA will schedule loads for the worst scenario following the two objectives by varying the value of β (see Section 6.3.6).

- *Analysis of the NGA with DER*. Finally, in the last simulations, the environment will also incorporate an storage system together with local PV electricity generator. The storage system is only able to charge the battery from the local surplus of generated electricity. Thus, it can be possible to defer the consumption of previously generated energy over time. In this experiment, NGA schedules the deferrable loads trying to reduce the variability of the grid and consuming as much as possible from the local sources. Those effects are analyzed varying the PV_p and the storage capacity (see Section 6.3.7)

In order to perform this analysis, the environment has been set following the elements configuration of Section 6.3.1, and all the different figures of merit used in the analysis are described in Section 6.3.2.

6.3.1 Environment configuration

For all the simulations, the aggregated consumption profile used corresponds to the one of the Spanish peninsular grid during 2015 according to official data provided by the grid operator Red Eléctrica de España (REE)[2]. The grid is divided in nodes whose local consumption is created by the virtual user already described in Section 6.1.1. Although the simulator allows creating as many nodes per lines as possible, the computation resources have limited the number of nodes that can be simulated. Thus, the number of nodes in which the grid is divided has been limited to 600 which are enough to study the behavior of the NGA.

Each node is composed by a virtual user in charge of the consumption, a PV generator and an storage system. The last two elements would be included in the simulations depending on the requirements of the experiment. In this case, the environment has been escalated to the mentioned number of nodes and each node represents more than one facility. So each node consists of the aggregation of various facilities composed by the same elements (consumption, PV generation and storage system).

The virtual user is in charge of creating the different loads of the node. All the loads created by the virtual user are deferrable and their $t_{i,j}^{act}$ is assigned following the local consumption pattern. In addition, this user will create the loads randomly and they can be created at any time during the simulation. If they are not scheduled by any controller, they will be executed at the creation time of the virtual user. For the simulations the length of the time step used is of 1 min. In the absence of a controller, each virtual user is going to follow the aggregated consumption pattern provided.

Nevertheless, each virtual user has a number of deferrable loads that can be created per period of time to simulate the same amount of energy that a local facility can consume. The virtual user has a window of time to set its deferrable loads of 1 day or 1440 min. Each deferrable load consists of an energy packet of a power amplitude of $P = 50\,\mathrm{MW}$ and a duration of $\Delta t = 60\,\mathrm{min}$. So, for the simulated grid to be equal to the real one, it is necessary that the energy consumed in both of them is the same. The Spanish grid in 2015 has a total energy consumption of 248 TWh, so the energy consumed per node in the simulator will be equal to 414 GWh. Finally, a node possesses around 8300 deferrable loads per year to simulate the behavior of the Spanish grid based on the energy consumed per node and the amplitude of the energy packets.

With respect to the local PV generation, the data used corresponds to the total predicted and measured PV generation of the Spanish grid for the year 2015[3]. The shape of the generation is extracted directly from this data. In order to incorporate

[2] *Spanish peninsular grid measurements, source:* R.E.E. e-sios

[3] *PV generation forecasts and measurements, source:* R.E.E. e-sios

inside the simulation, the data is normalized by the maximum generation of the year. The reason is that the amount of generated power varies in order to analyze the effect of different percentages of PV_p. For a $PV_p = 100\,\%$, the maximum nominal power of the generator is 240 MWp per node, such that the sum of the generation of all nodes is 144 GWp. This maximum nominal power is selected for the generator to produce the same amount of energy consumed in Spain in 2015, i.e. around 248 TWh.

The last element included in the simulations is an storage system per node. This simulated system consists of a battery inverter, a lead-acid battery and a battery controller. The capacity of the battery is modified in order to study its effects in the variability of the grid. Thus, the capacity of the battery is varied from the absence of capacity to 2 days of autonomy. The inverter parameters are adjusted to the battery and the controller will only charge the battery with the surplus of PV electricity. Hence, the function of the battery controller is double, maximizing the self-consumption without compromising the battery lifetime avoiding its over-charge and over-discharge situations.

6.3.2 Evaluation process

Each experiment is executed during 13 months or 568800 min. The first month is used by the algorithm to stabilize and the evaluation is done during the remaining time. NGA has been built by following two objectives. Thus, different figures of merit are needed to evaluate each part of the algorithm. In order to evaluate the smoothness of the aggregated consumption, the method selected analyzes its waveform. Whereas for the analysis of the second objective, a figure that relates the energy consumed with the one produced is required. In particular:

- *Analysis of the aggregated consumption variability.* It has been conducted primarily with the crest factor figure of merit (C_f), already used previously (see Sections 4.4 and 5.3). The mathematical expression is contained in Equation 4.31 and as a reminder, C_f is a relationship of the maximum of a signal with respect to its effective value during a period of time.
  For those experiments in which there is no local generation, the load factor (L_f) and demand factor (D_f) are also used as in the evaluation process of Section 5.3. The mathematical expression of the L_f is included in Equation 5.25 and for D_f is in Equation 5.26. Both of them are figures of merit used not only to analyze the form of the aggregated consumption but also the use that the different nodes of the grid made of the resource. In all the cases, the best value is 1 and they are evaluated for different periods of time, specifically: i) daily, ii) weekly, iii) monthly and iv) annually. The 3 evaluation parameters are averaged for this 4 periods of time as in Equation 5.27.

- *Analysis of the self-consumption.* In order to evaluate the second part of the NGA, it is necessary to define a new figure of merit. This figure should assess the amount of energy from the local energy resources that feed the consumption. Thus, the self-consumption factor (ξ) is defined as the fraction of the electrical energy consumed by the loads which is only supplied by the local generation sources (Castillo-Cagigal et al., 2011b,a). The self-consumption factor has two variants depending on the term used to normalize it:

  - *Consumption normalization.* It is defined as the fraction of load that is supplied (directly and indirectly) from the PV system. The mathematical expression is collected in Equation 6.8.

$$\xi_L = \frac{E_{DER}}{E_L} = \frac{E_{PV,L} + E_{Bat,L}}{E_L} \tag{6.8}$$

    where E_L is the total energy consumed by the loads and E_{DER} is the electrical energy generated locally which in this case is divided in: $E_{PV,L}$, the energy directly supplied by the PV generator to the loads, and $E_{Bat,L}$, the PV origin energy supplied by the storage system to the loads. Notice that depending on the environment configuration, the term $E_{Bat,L}$ could be zero and it represents the deferred use of the PV generated electricity in time.

– *Generation normalization.* It is defined as the fraction of generated electricity that is used to supply the loads and it is defined as

$$\xi_G = \frac{E_{DER}}{E_G} = \frac{E_{PV,L} + E_{Bat,L}}{E_G} \tag{6.9}$$

where E_G is the total generated electricity locally by the PV system.

ξ_L expresses the independence of the user over the grid while ξ_G represents the use of the local generated electricity to meet the demand of a user. Since the direct and indirect local use of PV production ultimately depends on the demand, it can be concluded that $\xi_L \in [0, 1]$ and $\xi_G \in [0, 1]$. However, the meaning of each factor is different. A $\xi_L = 0$ means that the node has no available local generation, while a value of $\xi_L = 1$ means that the energy consumed by the node is totally supplied by the local energy generation sources. On the other hand, $\xi_G = 0$ means that all the generated electricity goes to the grid and the user does not self-consume anything, and a $\xi_G = 1$ means that all the generated electricity supplies the local demand. It should be noted that both proposed factors could be used in different time frames. In addition, because they are normalized, comparisons between the ξ of PV systems with different sizes and loads are possible. However, only the ξ_L is used as the assessment factor for the simulations with local DER. The reason is that during the simulations, the energy consumed by the nodes is fixed and the DER sizes are varied in order to quantify their effects over the consumption.

6.3.3 Direct load control

In this first experiment, the EDeNC response is going to be tested in a real environment. The environment is composed only by consumption nodes, there is no local generation. Each node is equipped with a virtual user and the number of controllers varies depending on the L_C, the fraction of total consumption that is controllable. The consumption is composed by two different types of loads: deferrable, to create an aggregated consumption similar to one given, and elastic, to evaluate the controllable capabilities of the users.

Virtual users create deferrable loads following the local consumption pattern of the Spanish grid, but the EDeNCs are not scheduling $t^{act}_{i,j}$ of the loads. Thus, the deferrable loads are going to be executed when they are created. On the other hand, those nodes equipped with an EDeNC will use elastic loads in order to smooth the aggregated consumption. The EDeNCs modify directly the consumed power of the elastic loads. Thus, the power that these loads consume is given by multiplying the normalized output of the EDeNC by the maximum power that they can consume. In addition, the response of the controllers is executed in real time without taking into account the one week history information to elaborate it.

During the experiment, L_C is the percentage of nodes that is controllable. Each simulation used a different percentage of L_C and varies from 0 % to 100 %. As the deferrable loads of the virtual users are assigned randomly, each simulation is repeated for 30 different seeds per percentage of L_C. With this number of seeds, the dispersion for each realization is observed in order to validate statistically the experiment. The results of this experiment are represented in Figure 6.8.

Each panel of Figure 6.8 represents the median of each parameter for each of the 30 seeds used to evaluate each percentage of L_C. The deviation per simulation is also represented in shadow grey, but the $1st$ and the $3rd$ quartile are too close to the median so it is negligible. Thus, the EDeNCs achieve a very similar solution for all the seeds. In general, from the results of Figure 6.8, it can be observed that the tendency

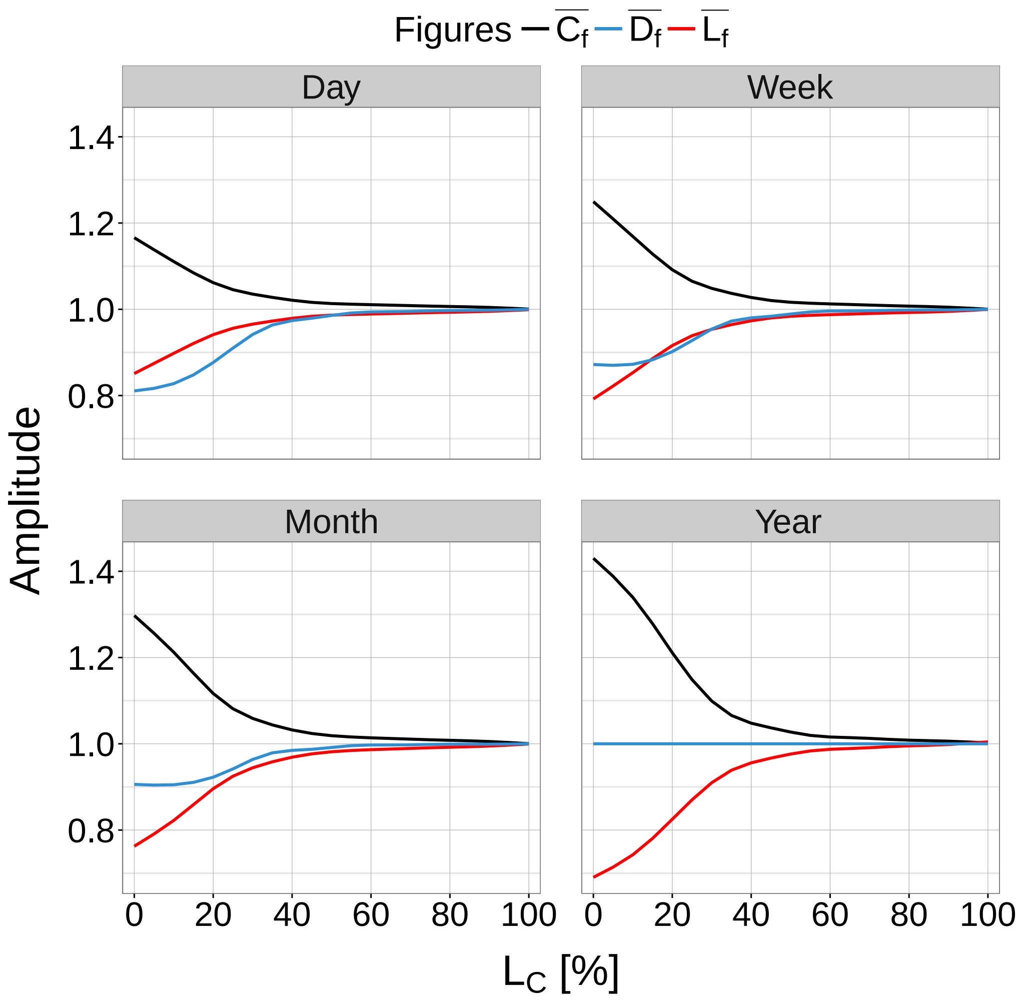


Figure 6.8: Evaluation of the EDeNC to control directly the power of elastic loads in real time. In black $\overline{C_f}$ per period of time, in blue $\overline{D_f}$ per period of time and in red $\overline{L_f}$ per period of time.

for all the three parameters in all periods of evaluation tends to the best value of each of them, 1, thus achieving the desired outcome (aggregated curve smoothed).

The first parameter to analyze is the $\overline{C_f}$. For no controllable capacity in the environment, the median $\overline{C_f}$ obtained per period are: 1.16 daily, 1.24 weekly, 1.29 monthly and 1.43 annually. With respect to the trend of the $\overline{C_f}$ value, it decreases as the L_C grows, but it can be observed in Figure 6.8 that there is a change of slope for $L_C = 30\,\%$. $\overline{C_f}$ decreases rapidly for L_C values less than this point and then decays slowly for bigger values of L_C until it reaches its minimum at $L_C = 100\,\%$. At a $L_C = 30\,\%$, the $\overline{C_f}$ value per period has decreased in 12.59 % daily, 19.13 % weekly, 22.50 % monthly and 30.58 % annually with respect to the absence of controllability capacity, $L_C = 0\,\%$. Finally, $\overline{C_f}$ reaches its minimum at the end of the experiment for $L_C = 100\,\%$. Thus, the $\overline{C_f}$ suffers a decrease in percentage of 16.50 % daily, 24.87 % weekly, 29.66 % monthly and 42.71 % annually comparing the starting $L_C = 0\,\%$ and ending $L_C = 100\,\%$. Thus, the inclusion of controllable elements in the environment enhances the grid status since its variability is reduced. Table 6.1 shows the results for all controllable capacities

L_C[%]	Day			Week			Month			Year		
	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$
0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	4.93	5.18	2.30	6.90	7.14	0.25	7.05	7.33	0.07	6.94	7.28	0.00
20	9.76	9.52	7.40	14.41	13.47	3.23	16.14	14.83	1.56	17.85	16.18	0.00
30	**12.59**	**11.82**	**14.21**	**19.13**	**16.89**	**8.99**	**22.50**	**19.25**	**6.25**	**30.58**	**24.42**	**0.00**
40	14.17	13.05	16.69	21.57	18.57	11.01	25.63	21.27	7.90	36.24	27.69	0.00
50	15.00	13.68	17.93	22.86	19.42	12.04	27.28	22.28	8.74	39.42	29.40	0.00
60	15.31	13.91	18.43	23.34	19.73	12.47	27.88	22.65	9.11	40.68	30.06	0.00
70	15.55	14.09	18.63	23.66	19.94	12.63	28.29	22.90	9.25	41.33	30.40	0.00
80	15.79	14.26	18.74	23.96	20.14	12.73	28.61	23.09	9.31	41.80	30.66	0.00
90	16.04	14.45	18.83	24.29	20.35	12.79	29.00	23.32	9.35	42.20	30.90	0.00
100	**16.50**	**14.79**	**18.98**	**24.87**	**20.73**	**12.88**	**29.66**	**23.69**	**9.41**	**42.71**	**31.32**	**0.00**

Table 6.1: Comparison of $\overline{C_f}$, $\overline{L_f}$ and $\overline{D_f}$ for all the periods of time and all controllable capacities in the direct control of the loads. In bold, the results discussed throughout the text are highlighted.

The second evaluation parameter is $\overline{L_f}$, the closer $\overline{L_f}$ is to 1, the flatter the aggregated consumption is. Thus, it is used as a measure of how well the grid responds to meet the demand. For all evaluation periods the trend is positive. So as L_C increases, $\overline{L_f}$ grows until it reaches its maximum for $L_C = 100\,\%$. In case L_C is non-existent, the median per period of the $\overline{L_f}$ are: 0.8515 daily, 0.7927 weekly, 0.7628 monthly and 0.6901 annually. As in the value of the $\overline{C_f}$, there is a change in the growing trend at $L_C = 30\,\%$. Prior to this point, $\overline{L_f}$ grows rapidly until it reaches that curvature point in which its growth rate decreases and the growth of $\overline{L_f}$ continues slowly until it reaches its maximum value of 1. At $L_C = 30\,\%$, $\overline{L_f}$ has an improvement with respect to the absence of controllable capacity of, 11.82 % daily, 16.89 % weekly, 19.25 % monthly and 24.42 % annually. At last, the increase of $\overline{L_f}$ in a grid with total control over the loads with respect to the one with absence of control is of 14.79 % daily, 20.73 % weekly, 23.69 % monthly and 31.32 % annually. As a result, it can be concluded that the inclusion of controllable elements within the grid improves the utilization of the available resources increasing its occupation by flattening the aggregated consumption. All the results for this factor are gathered in Table 6.1.

The last parameter of the evaluation is $\overline{D_f}$ which measures the utilization of a grid measuring the utilization of the maximum resources of the grid. Thus, for the annual period of evaluation, the result of this factor is always the same and equal to 1. The reason is that the aggregated consumption always reaches the maximum capacity of the system for the annual evaluation period. Figure 6.8 shows a sigmoid tendency for the $\overline{D_f}$ growth in the remaining periods. In the absence of controllers, the median per evaluation period is: 0.8102 daily, 0.8711 weekly and 0.9058 monthly. In this case, the value of $\overline{D_f}$ grows slowly until it reaches $L_C = 15\,\%$, then grows fast until it reaches $L_C = 30\,\%$ and finally grows slowly until it reaches its maximum at $L_C = 100\,\%$. So for the change of tendency point of $L_C = 30\,\%$, the perceptual increase per period is 14.21 % daily, 8.99 % weekly and 6.25 % monthly. Finally, the maximum $\overline{D_f}$ achieved was found at the end of the experiment for $L_C = 100\,\%$, which mean an increase of 18.88 % daily, 12.81 % weekly and 9.37 % monthly with respect to the absence of control in the environment. For all L_C, the results of $\overline{D_f}$ are in Table 6.1.

In view of these results, the inclusion of controllable loads enhances the status of the grid, reducing its variability and improving the use of the available resources. In addition, it is not necessary that the controllable loads exceed in number to the non-controllable one, but with a 30 % of the L_C, the results are quite visible within the grid. However, in this experiment the EDeNCs have total access to the power of the loads because they are elastic ones. So, it is necessary to move a step closer to the reality since the loads that allow accessing them and changing directly their power are few.

6.3.4 Random deferrable loads

Once the EDeNC response has been tested successfully, it is necessary to use them in a closer environment to the reality. In this experiment, the smoothing of the aggregated consumption is tested by using the EDeNC part of the NGA. As in the experiment of Section 6.3.3, the grid is composed only by consuming nodes and there is no DER inside them. However, the type of loads that compose the node consumption consists only of deferrable loads. Then, each node is equipped with a virtual user and the control capacity varies depending on the percentage of L_C.

The virtual users inside the nodes are the ones in charge of creating the deferrable loads using as consumption pattern the Spanish aggregated consumption. Depending on L_C, NGA is going to schedule the deferrable loads created by the virtual user. So for those nodes without NGA, their loads are executed in their creation time. As the output of the EDeNC is normalized, it can be used directly for the assignment of $t^{act}_{i,j}$. The execution of the loads is in real time, but the actions to schedule them are taken one day in advance in order to assign $t^{act}_{i,j}$.

The grid is formed by 600 nodes from which an L_C percentage of nodes are able to actuate over the scheduling time of the deferrable loads through the use of NGA. During the experiment the percentage of L_C is varied from 0 % to 100 %, i.e. from the absence of load control in the environment to its total control. As the loads are assigned randomly by the virtual users, each simulation was executed 30 times with different seeds. So, the necessity of using various seeds consists of evaluating the repeatability and dispersion of the solutions found by the algorithm for different load collocations in the times axis. This number of seeds is enough to validate statistically the experiment. Figure 6.9 shows the results of this experiment.

Each panel of Figure 6.9 represents a period of evaluation, from daily to annually. In each panel, the median of the average performance of each coefficient is represented in the period of evaluation for the 30 seeds in which each simulation is composed. Moreover, the deviation of each point of the median is represented in shadow grey. However, the distance of the $1st$ and the $3rd$ quartile with respect to the median is so small that it is negligible. Thus, NGA found a similar solution per each configuration of the experiment. As the previous experiment results, in Figure 6.9 can be observed that all coefficients reach the best value at the end of the simulation and the tendencies of each parameter are not as pronounced as in Figure 6.8.

With respect to the waveform of the aggregated consumption, the $\overline{C_f}$ is represented in black in Figure 6.8. In case of no control over the activation of the deferrable loads, the value of $\overline{C_f}$ per period is of: 1.16 daily, 1.24 weekly, 1.28 monthly and 1.41 annually. These values are very similar to the ones of Section 6.3.3. But in general, for all the evaluations periods, the decreasing tendency of $\overline{C_f}$ in Figure 6.9 is slower than the one of Figure 6.8. The reason is that the direct control allows to locate the exact quantity to be in antiphase to the aggregated consumption. Thus, the discretization error of the consumption profile plus the prediction error of using the EDeNC response produce values above 1 for $\overline{C_f}$. In this case, the decreasing tendency is more linear than exponential as in the previous experiment. Therefore, at the middle of the simulation ($L_C = 50\,\%$), the values of $\overline{C_f}$ correspond to a decrease of the 5.83 % daily, the 7.35 % weekly, the 9.44 % monthly and the 11.90 % annually with respect to the absence of controllability capacity. At last, $\overline{C_f}$ reaches its minimum at $L_C = 100\,\%$. Thus, the $\overline{C_f}$ has decreased in percentage an amount of 7.45 % daily, 11.41 % weekly, 13.37 % monthly and 19.55 % annually with respect to the beginning of the experiment at $L_C = 0\,\%$. In spite of the results being far from the unity in all periods, the inclusion of controllable elements in the grid flattens the

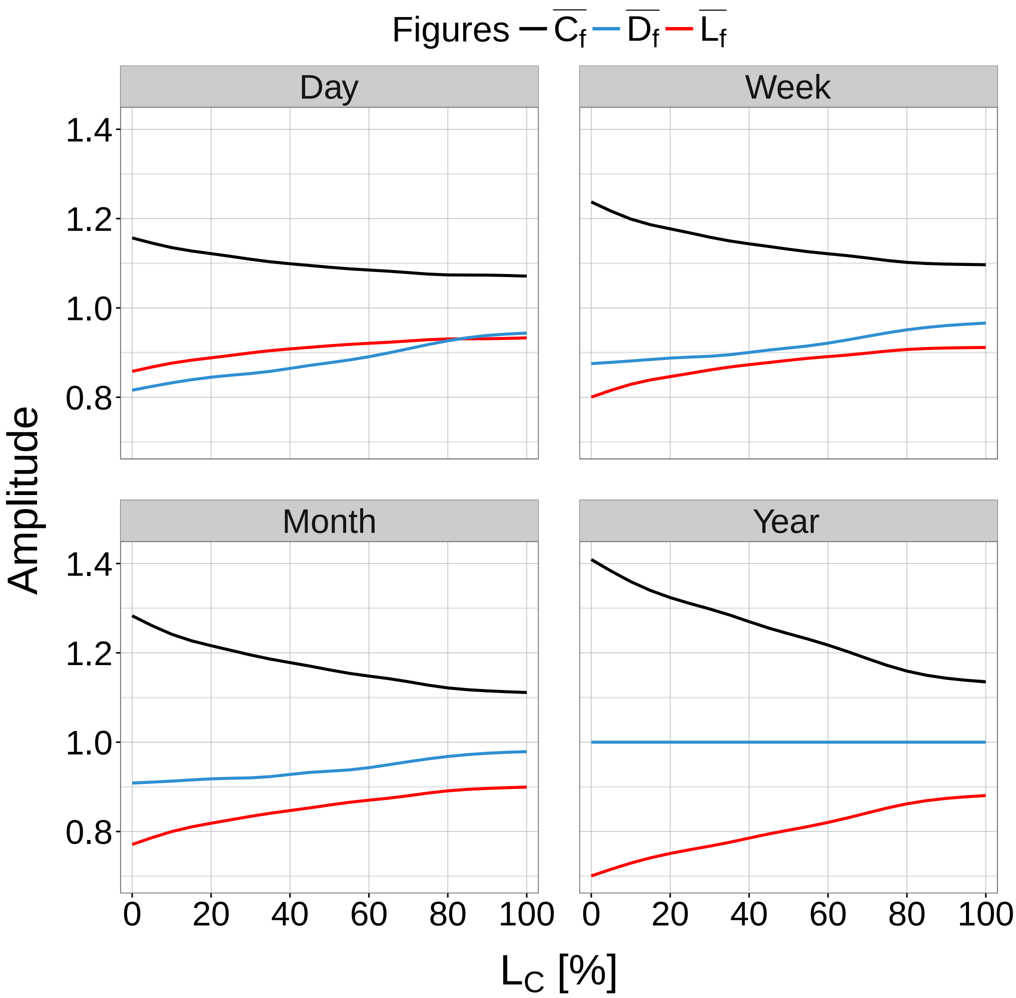


Figure 6.9: Experimental results of the NGA in order to smooth the aggregated consumption assigning the $t_{i,j}^{act}$ of the deferrable loads. In black the $\overline{C_f}$ per period of time, in blue the $\overline{D_f}$ per period of time and in red the $\overline{L_f}$ per period of time.

form of the demand curve. The rest of the values for all fractions of controllable loads are gathered in Table 6.2.

The next factor to analyze is $\overline{L_f}$ in charge of measuring the average occupation of the grid with respect to the available resources. As in the case of $\overline{C_f}$, the tendency of $\overline{L_f}$ is positive but slower compared to the one of Figure 6.8 for the same coefficient. The reason of this behavior is the same explained for the $\overline{C_f}$. The tendency in all periods of evaluation tends to be almost linear. In the absence of controllers to schedule the loads, the median value of $\overline{L_f}$ per period of evaluation is 0.85 daily, 0.79 weekly, 0.77 monthly and 0.69 annually. The value of the $\overline{L_f}$ increases as the L_C grows. Thus, at the middle of the experiment for $L_C = 50\,\%$, these values mean an increase in percentage of $\overline{L_f}$ about 6.86 % daily, 10.43 % weekly, 11.46 % monthly and 14.80 % annually with respect to $L_C = 0\,\%$. Then, $\overline{L_f}$ continues growing and getting closer to 1 until the best values per period are reached at the end of the experiment. Therefore, in a scenario with total control over the deferrable loads, it means an increase of $\overline{L_f}$ around 8.81 % daily, 13.97 % weekly, 16.63 % monthly and

L_C[%]	Day			Week			Month			Year		
	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$	$\nabla\overline{C_f}$	$\Delta\overline{L_f}$	$\Delta\overline{D_f}$
0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	1.88	2.16	1.71	3.18	3.69	0.22	3.26	3.77	0.06	3.35	3.98	0.00
20	3.07	3.55	3.63	4.94	5.84	1.17	5.38	6.31	0.95	6.1	7.25	0.00
30	4.19	4.89	4.19	6.50	7.75	1.56	6.88	8.23	1.33	7.63	9.27	0.00
40	5.00	5.86	6.00	7.61	9.14	2.80	8.20	9.90	2.23	10.15	12.46	0.00
50	**5.83**	**6.86**	**7.35**	**8.64**	**10.43**	**3.82**	**9.44**	**11.46**	**2.77**	**11.90**	**14.80**	**0.00**
60	6.35	7.49	8.93	9.52	11.53	5.08	10.54	12.91	3.44	13.42	16.87	0.00
70	6.80	8.02	11.43	10.24	12.45	7.12	11.59	14.24	5.52	15.94	20.41	0.00
80	7.34	8.67	13.16	11.17	13.65	8.29	12.72	15.76	6.32	17.49	22.71	0.00
90	7.17	8.48	15.15	11.31	13.85	9.67	13.18	16.39	7.34	18.78	24.68	0.00
100	**7.45**	**8.81**	**15.75**	**11.41**	**13.97**	**10.44**	**13.37**	**16.63**	**7.99**	**19.55**	**25.87**	**0.00**

Table 6.2: Comparison of $\overline{C_f}$, $\overline{L_f}$ and $\overline{D_f}$ for all the periods of time and all controllable capacities while scheduling random deferrable loads. In bold, the results discussed throughout the text are highlighted.

25.87 % annually with respect to the case with no control over them. In the end, using NGA to schedule the deferrable loads allow increasing the performance of the grid through a better use of the environment reducing the distance with the maximum. All the increases of this factor are shown in Table 6.2.

Finally, the last factor evaluated is $\overline{D_f}$. This factor of the three analyzed is the one that gets a closer value to its maximum for all periods of evaluation. At the beginning of the simulation, in the absence of controllers, the value of $\overline{D_f}$ is around 0.81 daily, 0.87 weekly, 0.91 monthly and 1 annually. As in the previous factors, the growth tendency does not have a pronounced slope as in Figure 6.8. Moreover, this slope is almost linear and grows slowly. Thus, in the middle of the experiment for $L_C = 50\,\%$, the value of $\overline{D_f}$ has increased in percentage with the respect to the beginning of the experiment in a 7.35 % daily, 3.82 % weekly and 2.77 % monthly. Then, the value of the factor continues growing until it reaches its maximum at $L_C = 100\,\%$. So, the factor has grown a 15.75 % daily, 10.44 % weekly and 7.99 % monthly with respect to the configuration with no controllable demand.

In view of the results and as conclusion of this second experiment, the inclusion of controllable elements favors the enhancement of the grid status. The use of NGA to flatten the aggregated demand achieved its objective and respect the preferences of the user. Moreover, adding controllers for scheduling the deferrable demand allows improving the usage of the grid, closing the gap between its maximum and the average consumption. Up until now, the response of the EDeNC and the NGA to flatten the aggregated demand were tested. In the next experiments, the capacity of the algorithm to integrate the local PV DG is analyzed.

6.3.5 NGA with DG

The first two experiments have been centered in the objective of smoothing the aggregated consumption through the response of the EDeNCs. This experiment serves also to test the smoothing capabilities of the NGA when adding DG of PV origin to the grid. So, the properties of the NGA would help to integrate the PV DG, mitigating the effects that it possesses over the variability of the aggregated consumption. In this case, the node topology is a bit more complex than the one used previously. Now, each node is composed by a virtual user in charge of creating deferrable loads plus a PV generator that supplies the electricity needed by the local loads. Moreover, depending on the L_C percentage, the node will be using the NGA to set $t_{i,j}^{act}$ of the deferrable loads.

The virtual users implement as consumption pattern the peninsular Spanish aggregated consumption in order to activate the deferrable loads created by it. On the other hand, those nodes equipped with a controller use NGA in order to set the $t_{i,j}^{act}$ of the deferrable loads but following the objective of smoothing the aggregated consumption, i.e. $\beta = 0$. Thus, the consumption pattern followed to schedule the loads is only the forecast of the normalized output of the EDeNC. The number of controllable nodes will depend on the percentage of controllable load, L_C. The remaining non-controllable nodes will activate their deferrable loads at the moment that they are created.

The grid is composed by 600 nodes with the elements already described above, in which not only L_C varies but also the energy of the PV generators through the percentage of PV_p. During the experiment the percentage of both L_C and PV_p are varied from 0 % to 100 %. Thus, all the possible combinations of both parameters are scanned during the experiment, from the absence of controllers and generation to an scenario in which all the loads are controllable and the amount of PV electricity produced is the same as the total energy consumed. Then, the grid is scanned searching for the best combination of parameters that improves the current situation. The PV per node use the waveform of the total Spanish PV peninsular generation and it is scaled taking into account the number of nodes and the PV_p for each simulation. For this experiment, only the measured total Spanish PV peninsular generation was used as the NGA only uses the normalized output of the EDeNC to schedule the loads. In addition, for each combination of parameters, L_C and PV_p, the simulation is repeated for 30 different seeds as there is a random component for each simulation.

In this experiment the figure of merit used to evaluate the form of the aggregated consumption is the $\overline{C_f}$. The other two coefficients used up until now are strongly related with the $\overline{C_f}$. Therefore, in order to simplify the analysis, the average value of $\overline{C_f}$ per period of evaluation is the only coefficient analyzed. The results of this experiment can be observed in Figure 6.10. They are represented in the form of heat maps divided in 4 panels, each one representing one evaluation period, from daily to annually. The value of $\overline{C_f}$ is the median of its average performance in the period of evaluation for the 30 seeds in which the simulation is composed. As in previous experiments, the deviation with respect to the median is negligible (less than 0.1 %) so it is not represented.

Figure 6.10 shows that, in general, in the absence of DG the variability of the aggregated consumption decreases as the L_C increases its value. In addition, as the percentage of the PV_p increases, the value of the $\overline{C_f}$ is far from the unity. The reason is that the increase of local generation produces a higher variability between the peaks and the valleys as the consumption is fed locally and there is no exchange with the grid. However, the use of controllers favors the integration of the DG since $\overline{C_f}$ decreases as the L_C grows since NGA schedules the loads during the valleys and reduces the consumption during the peaks. In the case that there is no controllable capacity and no DG, the aggregated consumption presents a value of $\overline{C_f}$ per period of evaluation of: 1.16 daily, 1.23 weekly, 1.28 monthly and 1.41 annually. These values are similar to the ones of the real Spanish aggregated consumption. The best values of $\overline{C_f}$ are achieved for $L_C = 100\,\%$ and the absence of DG ($PV_p = 0\,\%$). And the values that $\overline{C_f}$ takes in this scenario per period of evaluation are 1.07 daily, 1.09 weekly, 1.11 monthly and 1.13 annually. This entails a decrease in the variability in the best possible scenario for the different periods of evaluation of 7.76 % daily, 11.38 % weekly, 13.28 % monthly and 19.85 % annually. On the other hand, in the worst case scenario, in which there is no controllability ($L_C = 0\,\%$) and the amount of electricity produced locally is the same as the energy consumed ($PV_p = 100\,\%$), the value of $\overline{C_f}$ per period of evaluation is equal to 1.66 daily, 1.77 weekly, 1.83 monthly and 2.05 annually. In this case, consuming the energy locally will favor the self-consumption but it is worsening the variability of the signal. Thus, the value of $\overline{C_f}$ suffers an increase per period of evaluation of 43.10 % daily, the 43.90 % weekly, the 42.96 % monthly and the 45.39 % annually comparing to the absence of controllers and local PV generation. For any other combination of L_C and PV_p, the values that

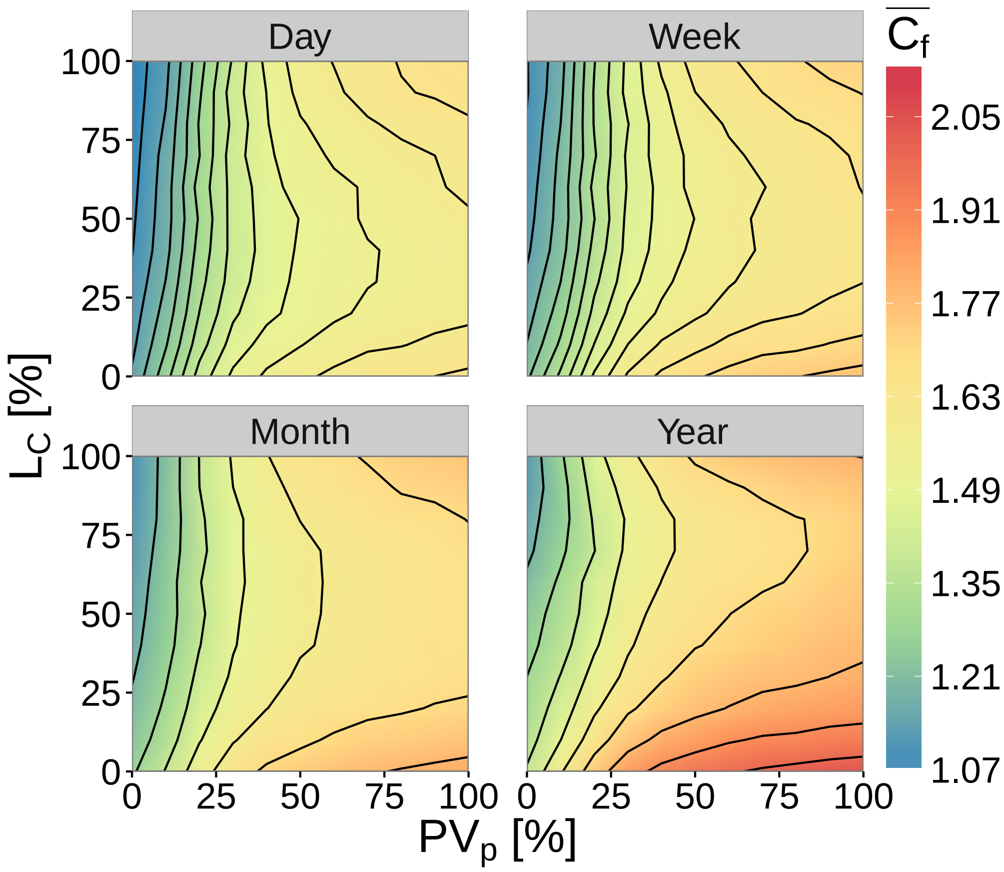


Figure 6.10: Experimental heat map results of $\overline{C_f}$ for the different combinations of controllable load capacity and photovoltaic electricity penetration using the Neural Grid Algorithm to smooth the aggregated consumption. In black, the contour lines that separate the different regions of achieved $\overline{C_f}$ values.

$\overline{C_f}$ can take are between them. Consequently, depending on the combination of the parameters, the local PV generation could be integrated inside the system without making worse the actual status of the system. In the best case, with $L_C = 100\,\%$, it could be integrated a $PV_p = 20\,\%$ without worsening the $\overline{C_f}$ values for $L_C = 0\,\%$ and $PV_p = 0\,\%$. Including controllable loads to the grid enhance its performance and make a better use of the local resources through their management with the NGA.

Up until now, the analysis has measured the effects of the aggregated consumption and the controllability of the grid. Now, as the PV DG appears in the environment, it is necessary to measure the use of the local available resources through ξ_L (see Equation 6.8) In this experiment, the value of the ξ_L is obtained by dividing the annual generated electricity that fed the local loads and the total annual energy consumed by them. Figure 6.11 shows the ξ_L results achieved with the local energy resources. The values showed in the heat map corresponds to the median value of the 30 realizations per configuration of the environment. In general, the higher the available local energy is, the greater the natural self-consumption is. Natural self-consumption means that without modifying the demand of the users ($L_C = 0\,\%$), the quantity of local demand that can be covered by local PV generation. As it can be observed, the maximum is reached for a $L_C = 100\,\%$ and $PV_p = 100\,\%$ and the ξ_L has a value of 50.18 %. This means that half of the local consumption would be supplied by the local generation. The reason is that the controllers, searching for smoothing the aggregated consumption, schedule the loads below the bell shaped PV

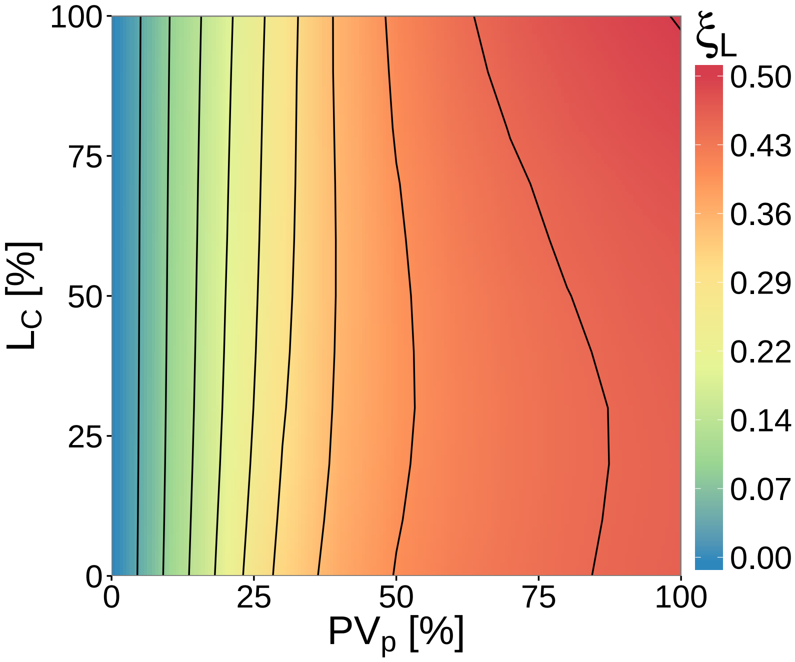


Figure 6.11: Experimental heat map results of the annual ξ_L for the different combinations of controllable load capacity and photovoltaic electricity penetration using the Neural Grid Algorithm to smooth the aggregated consumption. In black, the contour lines that separate the different regions of achieved ξ_L values.

electricity production in order to fill the valley created in the grid signal by supplying the loads locally. It can also be observed in Figure 6.11 that the increase of L_C helps increasing the value of ξ_L in those cases in which the local generation exceeds the local consumption, so that an electricity surplus would occur. The reason was explained before, NGA schedules more consumption below the generation hours in order to smooth the aggregated consumption curve. Thus, the contour lines of Figure 6.11 begin to bend from values greater than $PV_p > 35\,\%$. For a given PV_p, the value of ξ_L grows as L_C is greater.

In view of these results, the use of the NGA to schedule loads following the objective of smoothing the aggregated consumption not only serves to flatten it but also to integrate the local PV generation. In addition, it helps integrating in the best of the cases a 20 % of locally generated electricity without worsening the aggregated consumption of the grid ($L_C = 0\,\%$ and $PV_p = 0\,\%$). NGA also contributes to increase the self-consumption of the local generated electricity and in the best case half of the local consumed energy is supplied locally. Consequently, this supports that the distributed control approach of NGA serves to enhance not only the global objective of smoothing the aggregated consumption but also the local one of increasing the self-consumption of the local generation.

6.3.6 NGA following grid and PV

All the experiments described above are based on a common objective, specifically, to smooth the aggregated consumption in order to enhance the grid status. But, which are the consequences of adding the local generation to the algorithm as another objective? In the experiment of Section 6.3.5, the PV DG appears as an element of the grid and NGA only schedules deferrable loads following the EDeNC consumption profile. However, in this experiment, the second objective of the NGA to schedule deferrable loads is added. Thus, each controllable node is going to schedule loads following the EDeNC consumption profile and the forecast of the PV generation weighting the different objectives by the value of β. The node configuration is the same as in the experiment of Section 6.3.5.

In this case, the capacity of NGA to schedule loads following the antiphase aggregated consumption and the local generation profiles is tested together with the penetration of the PV local generation. Therefore, the entire population of nodes is going to be controllable, i.e. $L_C = 100\,\%$. This environment was chosen since the algorithm has full actuation capacity over the deferrable loads and its actions will affect directly the status of the grid. The virtual user is going to create the loads of the node and defines $\Delta t^u_{i,j}$ to schedule them with NGA respecting the user preferences. In order to schedule the loads, the followed consumption pattern is composed by the weighted sum of the EDeNC forecast output and the normalized PV generation forecast for the next day. Hence, depending on the β value of the algorithm, it will schedule the loads prioritizing one objective over the other.

As in previous simulations, the number of nodes used during the experiment is 600, all equipped with NGA to schedule the loads. The priority of one objective against the other is varied by changing the value of the β coefficient. In addition, the energy of the local generators will also vary in order to study the effects of different PV_p levels in the algorithm and the grid. Thus, during the experiment the value of β will be varied from 0 (schedule loads to smooth the demand curve) to 1 (schedule loads to maximize the self-consumption). Whereas the energy of the generators is varied using the percentage of the PV_p from $0\,\%$ to $100\,\%$. So, during the experiment all the possible combinations are scanned to identify the best ones that improve the performance of the grid and the local self-consumption. The PV generation used is the Spanish waveform denormalized to the amount of energy indicated by PV_p and the PV generation forecast from the grid operator. For each combination of β and PV_p, the experiment has been repeated for 30 seeds to check the repeatability of the experiment and its statistical validity.

As in the experiment of Section 6.3.5, the only factor analyzed is $\overline{C_f}$. This factor is used to evaluate the effects of the NGA and the PV local generation in the status of the grid. The results of this experiment can be observed in Figure 6.12. The $\overline{C_f}$ evaluation has been divided in 4 different periods of evaluation, from daily to annually. Then the results are presented in 4 panels, each one corresponding to a heat map for period of evaluation. The color scale is assigned depending on the value of $\overline{C_f}$ with respect to β and PV_p and it goes from the low values in blue to the maximum values in red. The value of the coefficient represented in Figure 6.12 is the median of the average performance per period of evaluation for the 30 seeds. As in previous experiments, the deviation with respect the median is negligible so it is not represented.

In general, the behavior observed in Figure 6.12 shows that the variability of the aggregated consumption increases as β and PV_p increase their value. In addition, as both parameters increase, the value of the $\overline{C_f}$ per period of evaluation is far from unity and even higher compared to the maximum of Figure 6.10. The reason is that as the value of β grows NGA will schedule more loads below the generation curve and as the local generator grows in size, the difference between off and on generation periods produces a bigger variability in the aggregated consumption. With respect to prioritizing one objective against the other, NGA achieves similar values of $\overline{C_f}$ for $\beta \in [0, 0.5]$ in all periods of evaluation. However, for $\beta > 0.5$, the algorithm gives greater weight to schedule the deferrable loads following the PV forecast which produces different values of $\overline{C_f}$ as PV_p grows. The best value of the $\overline{C_f}$ is obtained by the algorithm in the absence of DG ($PV_p = 0\,\%$) independent of the value of β since there is no local generation. These values are per period of evaluation equal to 1.07 daily, 1.09 weekly, 1.11 monthly and 1.13 annually. However, for values of $PV_p \neq 0\,\%$, $\overline{C_f}$ increases and it reaches the highest value for $PV_p = 100\,\%$. In this case, the values achieved by $\overline{C_f}$ per period of evaluation are 5.09 daily, 5.18 weekly, 5.23 monthly and 5.26 annually. Thus, the grid stability is in danger since the variability presented is

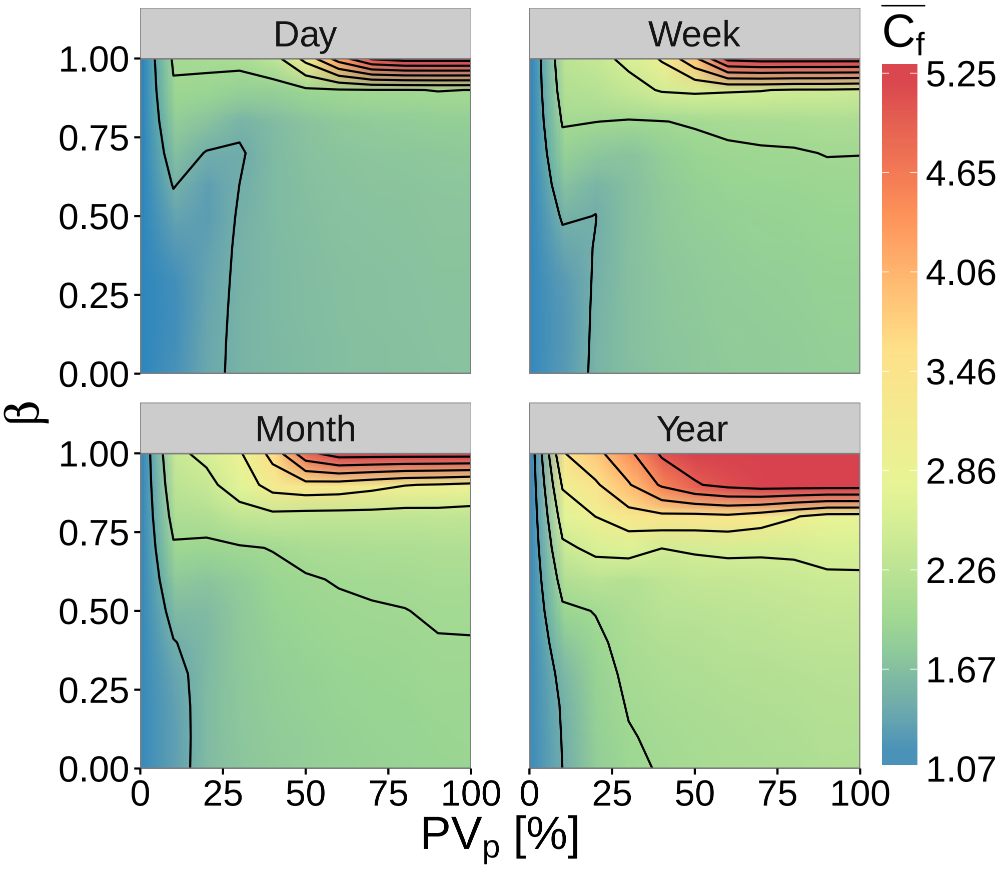


Figure 6.12: Experimental heat map results of $\overline{C_f}$ for the different values of photovoltaic electricity penetration using Neural Grid Algorithm weighting by β its two objectives: to smooth the aggregated consumption and to maximize the self-consumption. In black, the contour lines that separate the different regions of achieved $\overline{C_f}$ values.

too high. For any other combination of parameters, the value of $\overline{C_f}$ is between them. However, most of the results have lower values than the worst case. In case of $\beta = 0.5$ and $PV_p = 100\,\%$, the $\overline{C_f}$ value per period of evaluation is equal to 1.74 daily, 1.89 weekly, 1.99 monthly and 2.30 annually. These values mean an increase of percentage of 62.67 % daily, the 73.39 % weekly, the 79.27 % monthly and the 103.54 % annually. 60 % of the simulated environments are equal or below these values. In this case, NGA could integrate without worsening the grid $\overline{C_f}$ only a PV_p of 10 % for $\beta \geq 0.5$, whereas for $\beta < 0.5$ a 20 %.

On the other hand, the analysis is completed with the study of ξ_L. It is necessary to analyze not only the global effects that the variations of β and PV_p produce in the grid, but also the local effects in self-consumption of the local PV generation. ξ_L is calculated using the annual local energy generated that supplied the loads and the total energy consumed annually. In Figure 6.13, the results of the self-consumption of the local energy resources are presented. The value of each point of the heat map is the median value of ξ_L for the 30 realizations done by each configuration of the parameters. As in the results observed in Figure 6.11, the higher PV_p is, the higher the ξ_L value is. As can be observed, the increased β value helps to increase the value of ξ_L. This value grows until its maximum of $\xi_L = 62.31\,\%$ reached at a $\beta = 1.0$ and a $PV_p = 100\,\%$. Thus, comparing the maximum achieved in this experiment and the

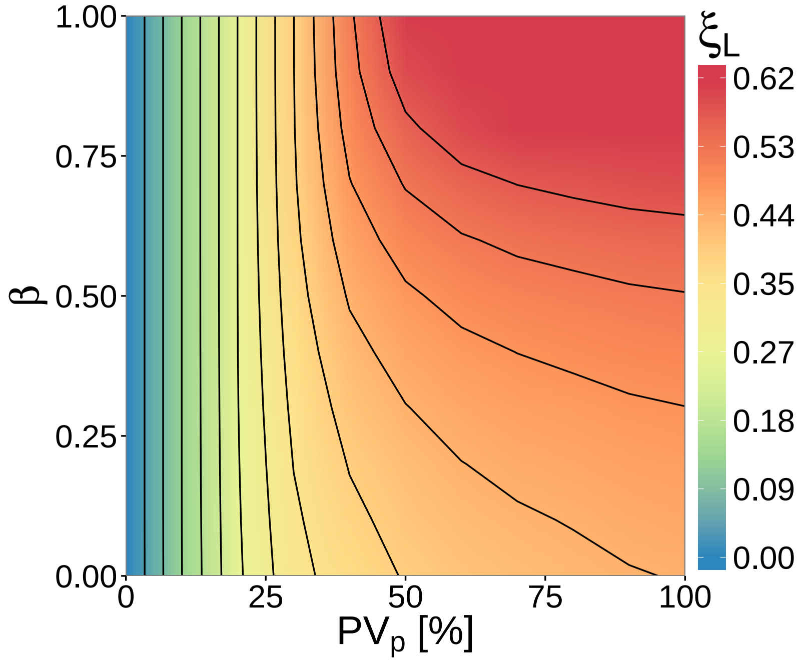


Figure 6.13: Experimental heat map results of the annual ξ_L for the different values of photovoltaic electricity penetration using the Neural Grid Algorithm weighting by β its two objectives: to smooth the aggregated consumption and to maximize the self-consumption. In black, the contour lines that separate the different regions of achieved ξ_L values.

one obtained in the experiment of Section 6.3.5, the increase of ξ_L was of 12.13 % by using the PV forecast to schedule the loads. In general, the increase of the β value helps increasing ξ_L. In Figure 6.13, for $PV_p > 25\,\%$ the surplus of generated PV electricity is better used as the value of β is increased. That is the reason why the contour lines present an exponential form from this value of PV_p. For previous values, the contour lines of ξ_L are linear and do not suffer any variation since there is no surplus of generated electricity and it is covered totally by the loads.

To sum up, the use of NGA to schedule the loads following the two objectives weighted by β does not only entail an increase of the self-consumption but also an increase of the variability of the aggregated consumption. So depending on the target, the value of β can be used to give more weight to one decision versus the other. In view of the results, it has also been stated that the two objectives of the NGA are conflicting since the increase of the self-consumption decreases the smoothing of the aggregated consumption. For high values of β and PV_p, $\overline{C_f}$ is far from the unity but more than the 60 % of the consumption is supplied locally. Thus, a better solution has to be found in order to increase the self-consumption without influencing negatively the aggregated consumption waveform.

6.3.7 NGA with DER

As already shown, adding the second objective to the NGA favors the self-consumption but the variability of the signal is increased. This fact has forced to rethink the use of the algorithm since in situations with high amount of local PV electricity generation, the algorithm produces non-desirable effects in the grid depending on the priority of the objectives. The increased variability occurs mainly due to the shape of the PV generation which presents a bell shape centered in daylight hours. This forces the algorithm to put a greater amount of deferrable loads during this period for a better use of local energy. So, there is a possibility to decorrelate the local PV generation to the consumption and it is the use of an storage system. This type of system allows not following the generation shape in order to maximize the self-consumption and distribute the local generated electricity along the time axis.

For this experiment, the use of the Distributed Energy Resources (DER) together with NGA is going to be tested. As the battery makes possible to differ the use of the local generated electricity in the time axis, the algorithm is going to schedule the deferrable loads with the objective of smoothing the aggregated self-consumption. Thus, it is not necessary to schedule the loads by following the local PV generation since the battery is going to be charged with the surplus of generation and the generated electricity is going to be used when required. Then, each node is going to be composed by a virtual user, a controller with the NGA, a PV generator and an storage system. As in Section 6.3.6, the totality of the nodes is going to be controllable since the flattened aggregated consumption is going to be altered by the use of the DER, being the worst case scenario. Thus, the virtual users only create the deferrable loads and establish a set of $\Delta t^u_{i,j}$ to schedule them with NGA. The deferrable loads are going to be scheduled with the consumption pattern of the EDeNC in order to smooth the aggregated consumption ($\beta = 0$).

In this experiment, the grid is configured with 600 nodes of the characteristics described above, in which the energy of the PV local generation and the nominal battery capacity (C_{bat}) are varied in order to study the enhancement of the grid and local energy resources. Thus, the percentage of PV_p is varied from 0 % to 100 %. The storage system of each node is composed by a battery with variable C_{bat} and an inverter to charge and discharge the battery, which is scaled depending on the C_{bat} and the size of the installation. The C_{bat} for this experiment is defined in days of autonomy which is the energy that the battery can store in order to supply the local consumption. Thus, C_{bat} varies from 0 to 2 days of autonomy which is enough to virtually isolate the node from the environment (Castillo-Cagigal et al., 2011a). The battery is only allowed to charge from the surplus of PV generation and it will be discharged until a safe SoC of 40 %. Finally, for each combination of parameters, PV_p and C_{bat}, the experiment is repeated 30 times in order to study the repeatability of the solution and the statistical validity of the experiment.

In order to study the waveform of the aggregated consumption, the parameter used is $\overline{C_f}$. The results of this experiment can be observed in Figure 6.14. $\overline{C_f}$ has been studied using 4 periods of evaluation, from daily to annually, and each one is represented in one panel of Figure 6.14. Each panel corresponds to a heat map whose scale depends on the value of $\overline{C_f}$ with respect to C_{bat} and PV_p. The color scale is assigned following the criteria of the closer to 1 the value of $\overline{C_f}$ is, the more blue the result is. On the other hand, the further the value of $\overline{C_f}$ is from 1, the more red the result is. The value represented in each heat map is the median of the average performance per period of evaluation for the 30 realizations per parameters combination. As in previous experiments, the deviation with respect the median is negligible so it is not represented.

In general, Figure 6.14 shows that the use of an storage system allows increasing the integration of the PV generation without increasing the grid crest factor as much as in the experiment of Section 6.3.6. Furthermore, the higher the values of C_{bat} and PV_p are, the greater is the value of $\overline{C_f}$. The reason is that each node is going to be more virtually isolated from the grid as C_{bat} increases and there is more available PV surplus to charge it. When the nodes need to consume from the grid it produces a high variability that it is translated in high values of $\overline{C_f}$. In this case, the inclusion of an storage system allows integrating more PV generation without worsening the $\overline{C_f}$. For a $L_C = 0$ %, the algorithm allows integrating a value of $PV_p = 35$ %. The values of $\overline{C_f}$ achieved per period of evaluation are 1.07 daily, 1.09 weekly, 1.11 monthly and 1.13 annually. On the other hand, the worst values were achieved for a $PV_p = 100$ % and a $C_{bat} = 2$ days of autonomy and they are per period of evaluation equal to 2.62 daily,

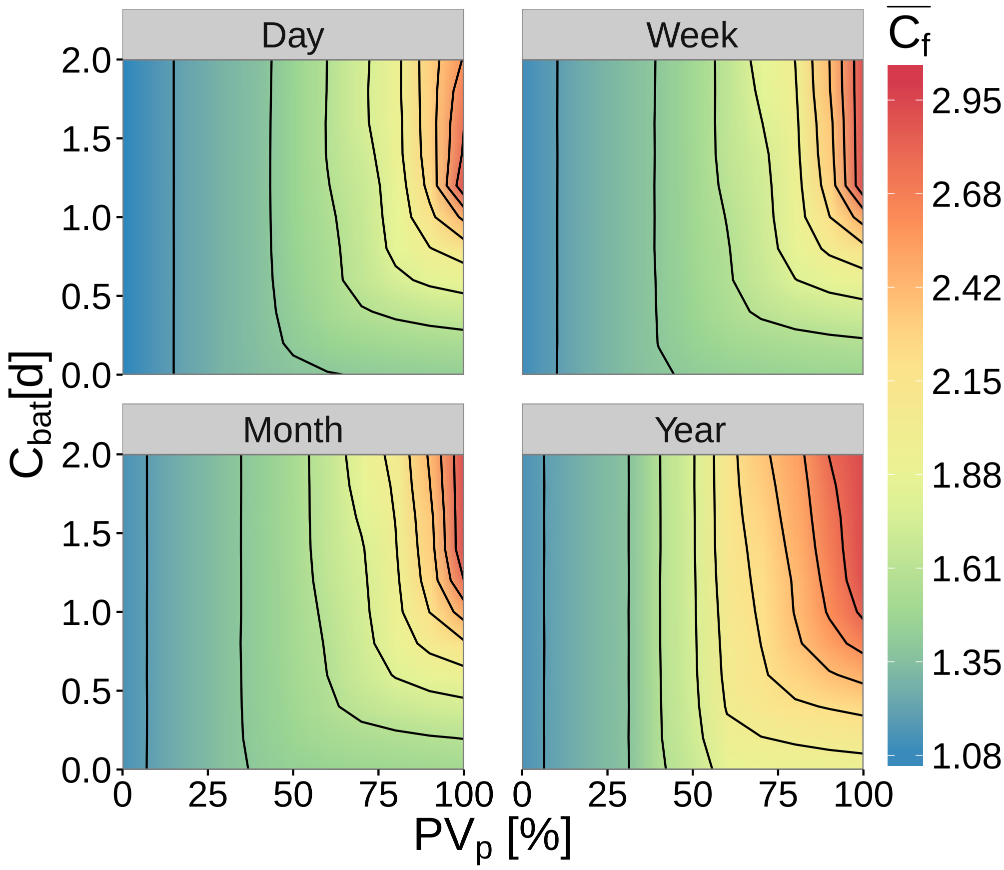


Figure 6.14: Experimental heat map results of $\overline{C_f}$ for the different combinations of nominal battery capacity and photovoltaic electricity penetration using the Neural Grid Algorithm to smooth the aggregated consumption. In black, the contour lines that separate the different regions of achieved $\overline{C_f}$ values.

2.71 weekly, 2.83 monthly and 2.97 annually. These values mean an increase of $\overline{C_f}$ with respect to the values no PV generation per period of evaluation of 144.86 % daily, 148.62 % weekly, 154.95 % monthly and 162.83 % annually. These values are still high but compared to the maximum achieved in the experiment of Section 6.3.6, the results are far better using an storage system to defer the use of generated electricity. In Figure 6.14, the contour lines change their form from a $PV_p \geq 50\,\%$. They have a form similar to exponential curves since as C_{bat} grows. The virtual isolation from the grid is bigger because the surplus of generated electricity is also bigger and consequently, the $\overline{C_f}$ values obtained are worse. However, at the same time, a battery with enough capacity ($C_{bat} \leq 0.5$ days of autonomy) to store only the excess of PV generation allows to relax the values of the $\overline{C_f}$ per period of evaluation.

The global effects on the aggregated consumption have been quantified up until now. However, it is necessary to analyze the local effects of adding an storage system plus the variation of the PV generation through ξ_L. Figure 6.15 shows the value of ξ_L achieved for the different combinations of the battery capacity and size of the PV generation. The value of ξ_L represented is the median of the 30 simulations done per each experiment configuration of both parameters. This factor is calculated using the annual generated PV electricity (directly from the PV generator or coming from the battery) supplying the local loads and the total energy consumed annually. The higher PV_p and C_{bat} are, the higher ξ_L is achieved. In addition, the use of the storage system helps increasing this factor since more local generation is been used to feed

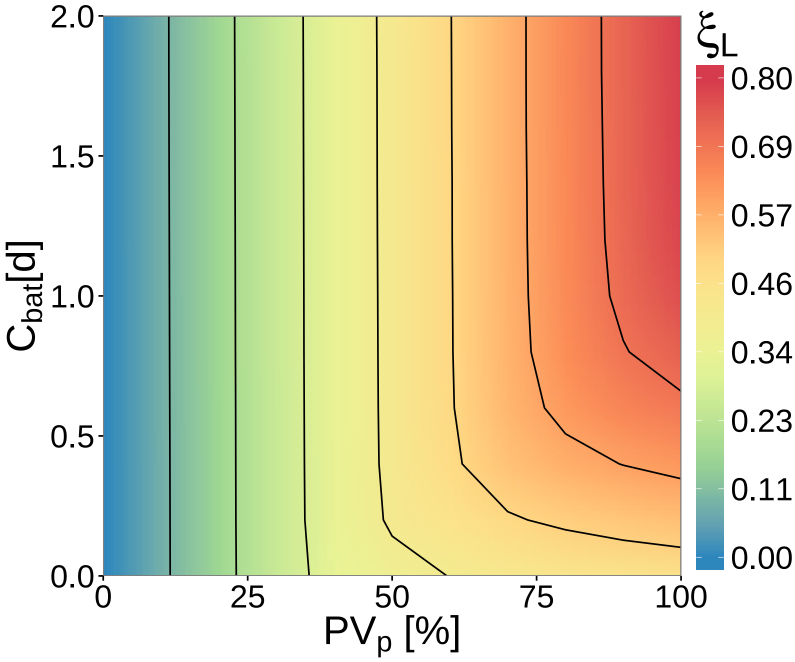


Figure 6.15: Experimental heat map results of the annual ξ_L for the different combinations of nominal battery capacity and photovoltaic electricity penetration using the Neural Grid Algorithm to smooth the aggregated consumption. In black, the contour lines that separate the different regions of achieved ξ_L values.

locally the consumption. The minimum value obtained is $\xi_L = 0\,\%$ which is achieved when no local generation is available. Whereas the maximum ξ_L is achieved at the end of the experiment for a $C_{bat} = 2$ days of autonomy and $PV_p = 100\,\%$ and it is equal to 79.98 %. Comparing to the maximums achieved in the experiments of Sections 6.3.5 and 6.3.6, it implies an increase of 29.80 % and 17.67 %, respectively. Thus, the storage system allows a better use of the local resources as shown by the contour lines of Figure 6.15. Furthermore, the use of an storage system does not only benefit the local self-consumption, but also globally, it allows integrating a higher percentage of PV generation and decreases the $\overline{C_f}$ of the grid, as if the loads were consuming all the local electricity generated at the same time that it is produced.

In conclusion, this experiment shows that adding an storage system to the node is a better option in order to increase the self-consumption of the local resources and not increasing in excess the variability of the aggregated consumption. The storage system allows consuming the PV electricity not only at generation hours, so that relatively low C_{bat} that absorbs the surplus of the PV enhance the system both locally and globally. In addition, even with higher values of PV_p, it has been reduced the effects seen in Section 6.3.6 but the variations introduced are still too high. However, the storage system allows bigger quantities of PV generation inside the grid, $PV_p \leq 35\,\%$, and it achieves that 80 % of the consumption can be fed locally.

6.4 Summary and discussion

In this Chapter the EDeNC response, developed in Chapters 4 and 5, was tested in a simulated environment which resembles a real grid. For this purpose, the

GPLV3.0 open simulator *GridSim*[4] was used as a tool to develop all the analysis and experiments. *GridSim* is a powerful tool that simulates the power flows of an electrical grid and models different elements inside the grid that resembles the real ones. In this case, the facilities used during the simulations were composed by three elements: consumption, local PV generation and an storage system. The consumption is formed by the three types loads: deferrable, non-deferrable and elastic. Thus, the direct application of the EDeNC is complex inside this environment. In addition, with the appearance of local energy sources, a new objective was posed in order to enhance its local use.

In this scenario, the proposed algorithm in this Thesis has been completed and validated according to two objectives: one global, smoothing the aggregated consumption of a grid, and one local, maximizing the self-consumption of the local energy resources. This proposed algorithm, called Neural Grid Algorithm (NGA), was developed based on the use of the response of the EDeNC and the PV DG that the facilities incorporate. In order to achieve the first objective, the NGA uses the properties of the EDeNC to reduce the variability of a signal, which corresponds perfectly to the objective of smoothing the aggregated consumption of a grid. On the other hand, the second objective of the NGA needs to include information relative to the local generation in order to decide when the facility should consume and take advantage of it. Therefore, the NGA incorporates the forecast of the day ahead for the local PV generator. Both objectives are weighted by a β coefficient in order to give different priority to the objectives.

NGA actuates over the demand of the facility based in these objectives. The EDeNC response is prepared to actuate directly over the power of the facility, however in a real environment a few consumptions can be modified directly. Hence, the NGA used the consumption profiles of both objectives in order to schedule the different loads of the user. Then, the consumption has been discretized in energy packages that represent the loads of a facility and depending on the grade of the controllable load capacity (L_C), the algorithm has more energy to defer following the two objectives. These energy packets are created by a virtual user which uses a consumption profile to schedule them in probability. It is the virtual user which establishes the amount of energy that it is deferrable, non-deferrable or elastic. Thus, the NGA would actuate over the activation time of the deferrable loads respecting the user preferences and scheduling them by the use of a probability density function based on the weighted sum of the two objectives of the algorithm.

Once the definition of the algorithm has been introduced, five different experiments were designed in order to test it. Each of them tests different aspects of the algorithm. The first one tests the direct use of the EDeNC response with elastic loads over the grid by varying L_C. The second one test the load scheduling capacities of the NGA in order to smooth the aggregated consumption by varying L_C. In the third one, the PV DG was added to the facilities to test the integration of the PV generation when the NGA only schedules loads based on the first objective. In this case, L_C and photovoltaic electricity penetration (PV_p) were varying. The fourth experiment studies the full capabilities of the NGA in order to schedule the loads based on the two objectives. In this case, L_C was equal to 100 % and β and PV_p were varied. Finally, the last experiment, following the results of the previous one, used an storage system together with the PV generation in order to enhance the grid status. In this experiment, the NGA was used to schedule loads with the objective of smoothing the demand since the properties of the storage systems were tested. L_C was equal to 100 % and C_{bat} and PV_p were varied. In the first two experiments, the analysis consisted of studying the average crest factor ($\overline{C_f}$), average load factor ($\overline{L_f}$) and average demand factor ($\overline{D_f}$) to quantify the effects on the grid. Whereas in the rest of the experiments $\overline{C_f}$ and self-consumption normalized by the consumption (ξ_L) were studied to analyze the effects on the grid and quantify the local use of the DER.

As a general remark, the inclusion of L_C with the NGA improves the grid status through the reduction of the variability of the aggregated consumption and also improves the use of available resources. In case of using elastic loads, it is needed

[4] Source: https://github.com/Robolabo/gridSim

a 30 % of L_C to achieve an annual reduction of the variability of 30 %. Whereas using the algorithm to recommend the user when is the best time to activate the deferrable loads, it is needed an $L_C = 50\,\%$ to at least reduce annually the variability in a 13 %. In addition, the NGA presents some errors due to the discretization of the consumption and the use of the forecast response of the EDeNC. Furthermore, the use of the NGA following only the first objective allows a better integration of the PV DG as L_C increases. But, it allows that for values of PV_p less than 20 % the grid status does not get worse and the maximum ξ_L achieved is of 50 %. At the same time, the NGA allows enhancing the self-consumption of the facility increasing the use of the local energy sources. With respect to the two objectives of the NGA, prioritizing the second objective produces a higher variability on the grid but at the same time higher values of ξ_L are achieved around 60 %. For values of $\beta \in [0, 0.5]$ the algorithm is able to maintain relatively similar variabilities in the grid and achieves a high use of the local resources. Finally, the inclusion of an storage system relaxes the constraints of the environment and allows consuming the generated electricity when it is needed. Thus, with an storage system, the PV_p achieved is of 35 % and at the same time reduces the variabilities achieved by scheduling loads following the NGA objectives. In addition, the use of the local energy resources is improved and reaches a value of the 80 %.

PART III

Conclusions

7 Conclusions and Future Works

"Stay hungry. Stay foolish" — Rashmi Bansal

Key features of a self-organized coordination algorithm to manage the power flows of an electrical grid have been presented in this Thesis. The aim of the Thesis was the application of Artificial Intelligence (AI) algorithms, specifically Recurrent Neural Networks (RNNs), to the distributed control of a grid with the presence of Distributed Energy Resources (DER). In this last Chapter, the main achievements and remarks of the Thesis are summarised, both in the computer science and energy fields. The remainder of the Chapter is as follows. Section 7.1 presents an overview of the developments, improvements and results obtained within the objective proposed in the present Thesis. Section 7.2 presents different application possibilities of the results of this Thesis and some proposals to develop in future research. Finally, Section 7.3 summarizes the main contributions of the author over the course of fulfilling this dissertation.

7.1 Conclusions

The present structure of the electrical grid has not suffered disruptive changes since its creation in the XIX century. However, its evolution has become essential as some factors are threatening its stability such as the growing demand, while others are being recently developed inside it without their proper integration such as Information and Communications Technology (ICT) or Distributed Generation (DG). The emergence of DG supposed a change of paradigm in the structure of the grid, since the generation is closer to the places where it is consumed, enhancing the local system status. Furthermore, the degree of connectivity between the different grid elements has been increased thanks to ICTs, being necessary to process all the data gathered in order to enhance the grid operation. The present Thesis sheds light on the problems arising from the management and operation of existing electrical grids and their evolution to what is considered the grid of the future or Smart Grid (SG).

The SG has arisen from the convergence of five key aspects that all the definitions have in common: i) the grid, ii) ICTs, iii) renewable energies, iv) Electrical Energy Storages (EESs) and v) Demand Side Management (DSM). In this Thesis, a definition of the SG has been suggested to explain the meaning of this huge concept and to understand what would the next generation of grids become. Thus, in the framework of the present Thesis, a SG is considered as an electricity network that uses the ICTs to coordinate the needs of all groups inside the system to operate all its parts as efficiently as possible, minimizing costs and environmental impacts while maximizing system reliability, resilience and stability. The SG presents benefits for all the members of the grid (security of supply, reliability, efficient, etc.), and also for the environment. SG is still at an early stage to be deployed and a lot of research has to be done before its final stage arises. However, it also faces a lot of barriers to be fully adopted, but the most worrying problem is the lack of investment to carry out the development of these technologies.

This Thesis consists of a first step towards the SG by linking and integrating the five key aspects for its development and deployment in the near future. To this end, an algorithm is proposed to process the tones of data that the ICTs gather in order to benefit all the parts of the grid. Greater operational efficiency, cost reductions and reduced risks are achieved. The current centralized structure of the grid does not favor the information processing as the data load is large enough to take decisions by the operators. Thus, the proposed algorithm follows a decentralized approximation in order to alleviate the data load of the centralized node. In this scenario, the decentralization was taken to the end of the grid, making the users participants in their decisions and being able to manage their power flows into a common objective, increasing the sustainability of the system, through DER and an efficient use of electricity generated. The proposed algorithm is based on DSM techniques combined with an automatic control of demand that helps to integrate DER (renewable energies and EES), which leads to an innovative concept called Active Demand Side Management (ADSM). This ADSM algorithm serves as the cohesive element that links all the elements that define the SG.

Among all the possibilities to implement an ADSM algorithm, the proposal of the Thesis is based on one of the famous algorithms coming from the field of the AI, the Artificial Neural Networks (ANNs). They have been studied since the 50s, but it was during the 80s that almost all the developments were done. Some problems were found during this time, but they were overcome and nowadays they enjoy of a fame well deserved thanks to their properties. As the ADSM algorithm is going to be used in large scale and changing environment, ANNs were selected for their properties of adaptivity, generalization, redundancy, fault tolerance and distributivity, among others. In addition, the modularity of ANN offers the possibility to divide a complex problem into easier tasks that help to solve the global problem. All the properties that the ANN possesses are similar to the ones that the future SG will provide to conventional power systems. There exist plenty of ANN types, but for the proposed algorithm there are two fundamental attributes that must be targeted, the time variable and the presence of memory. So, the type of ANNs selected is the RNNs, specifically the Continuous Time RNNs (CTRNNs) since it includes all those properties in its structure plus dynamic features that help in a changing environment.

The objective of the proposed ADSM algorithm is to enhance the performance of the grid. To this end, the local consumption is managed to reduce the variability of the aggregated consumption producing an smoothed signal and at the same time it also responds to local conditions. The ADSM strategy followed is applied locally but its effects impact globally in the system. Therefore, it is necessary to involve users in this process and the coordination between facilities and the grid should be performed, where a facility is owned by a particular consumer whose management depends on him and is composed by Photovoltaics (PV) generation, an storage system and consumption. However, the coordination process is done without any communication among the users. Thus, the only information available to coordinate the ensemble of users is the aggregated consumption of the grid. Data protection and anonymity is guaranteed following this approach. Consequently, the different users inside the grid are self-organized to contribute to smooth their aggregated consumption. To this extent, the CTRNN was used to control the local electric behavior taking into account the aggregated consumption of the electrical grid, local energy resources and users requirements.

In the implementation of the ADSM, it was crucial the use of the CTRNN because its signal processing and forecasting abilities make particularly interesting their use in a changing environment as the grid. In addition, the dynamic properties and the short-term memory of its structure helped to create an approximation to reduce the variability of a signal. The aggregated consumption is a complex signal which depends on the behavior of all users connected to the grid. However, this signal presents some characteristics that helps to identify an approach to flatten it. Among its features, it presents that it is continuous, periodic, bounded and differentiable of class C^1. Thus, taking into account these signal features, a CTRNN controller is built to cancel the derivative of the aggregated consumption achieving a constant aggregated signal. Consequently, the controllable users build a destructive interference through their consumption that it is in antiphase to the one corresponding to the non-controllable users and as a result, the aggregation of all the consumptions will generate a smooth aggregated consumption.

However, it was necessary to define the structure of the CTRNN to be able to cancel the derivative of the input aggregated signal coming from the environment. During the selection of the architecture some considerations were taken about its size since large CTRNNs are difficult to analyze and computationally slow. Thus, small structures were explored in order to be able to use them together with any technology inside any type of user facility. At the same time, a modular approximation was followed consisting of neural blocks where each one solves parts of the problem. In order to find the right structure of the neural controller, a second algorithm was used to tune its synaptic weights, the Genetic Algorithm (GA). During the search of the best structures, not only different parameters of the neural controller were adjusted but also an analysis of the best configuration of the GA was carried out in order to understand the influence of its parameters in the search of the best solution. The GA evaluated the capacity of the evolved structure to cancel the variability of the input signal through the extraction of its derivative and the consumption of an antiphase output. To this end, $388,800$ simulations were carried out for all the different neural structures. The best structure found was the one formed by an input layer of 2 neurons, a hidden layer of 4 neurons and an output layer of 1 neuron. The information goes feedforward between layers, each neuron presents a feedback loop with itself and the ones of the hidden layer present feedback loops among them. In addition, there is a feedback loop from the output to the input, so the network can store its last actuation. A post-evaluation was done to analyze the response of the evolved neural controller and for all the tested signals, a variability reduction of the input signals between the $60\,\%$ and the $16\,\%$ was achieved.

The evolved neural controller is the central block of the ADSM controller. However, its direct application in the grid could not be done since all users will consume the same and the grid is composed by an heterogeneous environment. Moreover, if all users try to consume the same power at the same time, greater variabilities of the aggregated consumption are reached. Thus, a coordination algorithm was implemented in order to produce collectively the antiphase consumption necessary to flatten the aggregated consumption. Inspired in the Blind Source Separation (BSS) techniques, the τ-Learning Algorithm (τLA) was used to regulate the output of the neural controller ensemble based on the last free parameter of the network, the time constant (τ_i). This parameter is related with the reaction speed of the neurons. The τLA was implemented following a reinforcement learning paradigm in which the minimization of the variability of the aggregated environment signal is used to change the value of this parameter. The use of the τLA implies that after the evolution, each evolved neural controller was developed to obtain its own output different from the majority of controllers. With this achievement, the birth of the Evo-Devo Neural Controller (EDeNC) arose to control the local demand of a user. The EDeNC could be used in two modes of operations: adapting to the environment with or without power constraints. In both cases, an smoothed aggregated environment signal is achieved from a neural distributed way.

Finally, the direct use of the EDeNC is not recommended as it actuates directly over the power of the user and there are a few loads that allows this behavior. In addition, local Distributed Energy Resources (DER) considerations are not taken into account in the EDeNC. Thus, the proposed algorithm, which is implemented as an ADSM controller, uses the response of the EDeNC and a local PV generation forecast to elaborate a consumption pattern of the user. This algorithm or Neural Grid Algorithm (NGA) has two objectives: one global, the smoothness of the aggregated consumption and one local, the maximization of the local energy resources use. Both objectives are weighted by a β coefficient to establish different priorities in the objectives. Then, the consumption pattern generated by the NGA is used to schedule the different deferrable loads that the user possesses. The algorithm was tested in a virtual environment for selected scenarios closer to reality. The data used in the simulations were the total PV generation forecast of one day ahead, the total PV generated and the aggregated consumption of the Spanish peninsular grid in 2015. As a general remark, the inclusion of the loads controllable fraction,

L_C, with the NGA improves the grid status through the reduction of the aggregated consumption variability plus a better use of the available energy resources. It is needed an $L_C = 50\,\%$ to at least reduce annually the variability in a 13 %. In addition, the use of the algorithm allows integrating the local PV generation for photovoltaic electricity penetration (PV_p) less than 20 % without worsening the status of the grid. The use of local storage allows a bigger integration ($PV_p \leq 35\,\%$) and a higher use of the local energy is achieved. As conclusion, the algorithm was able to adapt the controllable demand to the non-controllable one and thanks to its use, the NGA reduced the yearly variability of the aggregated consumption. Thus, the NGA meets the main objective sought in this Thesis.

To conclude, the use of the NGA developed in this Thesis entails an important contribution to the deployment of the future SG. With its implementation, some needs and barriers referred in Sections 2.2 and 2.3 were tackled. The benefits that introduced the use of the NGA are summarized as follows:

- The *efficiency* of the system is increased with the reduction of the peaks as less power is demanded instantaneously. The use of the NGA also helps to decrease the oversizing of the electric system in order to guarantee the supply. And it also helps the grid operators with their daily tasks such as the scheduling of the generation for the next day, integrating some forms of energy, etc.

- Increasing the *robustness* as the redundancy of infrastructures inside the grid is increased. As a distributed algorithm, if one of the facilities fails, the remaining facilities will adapt their consumption to continue smoothing the aggregated consumption. In this case, the distributed property of the algorithm is doubled, not only at the facility level but also the EDeNC presents this property inside the facility. In addition, the robustness is also increased since the essential information is sent from the nodes to the grid and vice versa. Compromising the information through the ICTs is more difficult due to the reduction of node communications.

- *Data privacy* is assured since no communication between the nodes is available with the algorithm.

- Reduced *computational load*. As all the decisions are made from the side of the user, the computational load of processing and making decisions is not as high as if a central node was in charge of doing all these operations. Moreover, the simplified structure of the EDeNC makes possible its execution in any current technology since the computational effort is not huge.

- Reduced *investments and costs* with the use of NGA since it adapts to the current status of the grid. Some modifications are still required such as the upgrade of the sensory platform in order to establish a double-sided communication. However, it is not necessary that each node possesses high computational power or any other sophistication, since only providing the form of the aggregated consumption will be enough to deploy the NGA and take benefit for all its features.

7.2 Future Work

This Thesis is the NGA mark I. Although a first approximation of the future SG was achieved with this algorithm, further development is required in order to improve its performance. In this Thesis, a combination of the Artificial Intelligence (AI) and the energy management fields were found. During the development of this Thesis, different research lines have been identified for future exploration that would be interesting to study.

The algorithm developed in this Thesis is still in a first version of its development. Furthermore, the NGA is not mature enough for its deployment at large scale in real grids, but it could be used in reduced environments, such as buildings or microgrids. However, some improvements should be made in order to use its final implementation in real environments. These improvements are listed as follows:

- *Improve the prediction horizon.* One of the problems found with the NGA to smooth the aggregated consumption is the scheduling of the loads with the real time operation of the EDeNC. The EDeNC was able to predict one step ahead to cancel the derivative of a signal and thanks to the periodicity of the signal it was able to achieve its purpose. However, the period of repeatability was of one week being too much time for the algorithm if some unexpected event happens. Thus, increasing the forecasting capabilities of the EDeNC would increase the performance of the algorithm and unexpected events would be taken into account. To do so, following the task partitioning approximation by dividing the problem into simpler parts, a specific prediction module could be added to the current form of the EDeNC. Hence, the EDeNC would not be modified and this new module could be as complex as it is required to elaborate the prediction of the aggregated consumption. On the other hand, another possibility would consist of the modification of the EDeNC structure by increasing the history and using more outputs corresponding to the different time steps of the forecast. In this way, it would be necessary that the structure of the EDeNC would follow a Nonlinear Autoregressive Moving Average with eXogenous inputs (NARMAX) approximation. Apart from these two types of approximations, a third way can be used to increase the prediction horizon. Changing the neural structure could also be helpful. For example, the use of Long Short Term Memory (LSTM) networks would help to increase the forecasting capabilities without compromising the neural structure. LSTM networks are a type of RNN which learn from experience to classify, process and predict time series when there are important long time lags of unknown size between events.

- *User consumption profiles.* In this Thesis, a virtual user was in charge of creating loads with respect to a consumption profile. However, only elastic and deferrable loads were used separately to show the characteristics of the EDeNC and NGA, respectively. A more complex consumption profile could be achieved if the three types of loads defined in this Thesis (non deferrable, deferrable and elastic) were used together and assigned by the virtual user. In this way, a more realistic demand profile per user should be simulated and the strategy of scheduling loads by the NGA should be reconsidered. The control capacity of the NGA should be increased in order to be able to manage the three types of loads added. For the non-deferrable loads, the algorithm has to ignore them since they are going to be executed when the virtual user needs them. The scheduling capacities of the NGA are still valid, however the algorithm has to take into account not only the deferrable energy packages but also the direct power load control over the elastic ones. In addition, the specifications of the elastic loads are different from the deferrable ones since each one presents different features: for example, the Electric Vehicle (EV) is a mobile load that can also be used as an storage system and the Heating, Ventilation and Air Conditioning (HVAC) loads maintain a comfort temperature ordered by the user. Deferrable loads could also increase their realistic form if the real consumption profiles of these loads were implemented. In this Thesis, only the end user consumption profile was used based on the residential sector. However, different types of users have to be integrated in the system to give more realism to the virtual grid environment, e.g. industrial or commercial users, as other special loads that the grid operators use.

- *Multi purpose algorithm.* The NGA presented in this Thesis is composed by more than one objective. However, both targets were antagonistic and the increase of one of them will imply the decrease of the other. A better integration of the two objectives was done with the addition of an storage system in the environment but it was not part of the algorithm. Therefore, adding the storage system as part of the algorithm could suppose a great asset as both objectives are better integrated and the algorithm achieves better performance. Moreover, storage systems are increasingly used due to their value in terms of flexibility, technological improvements of the different technologies, the appearance of new solutions from different companies and the emergence of the EV. The paradigm of the storage systems has been changing since the introduction of the EV because the storage system is not static anymore, it is mobile. The peculiarity of the EV is that it can be used as a load or as a storage system, always taking into account not only the energy requirements of the user but also their mobility capacity, as it can be connected in any part of the grid. For example, the user needs the vehicle to move from home to another place in which is going to do an activity, during those periods of time that the EV is not used, the NGA could use it to charge or discharge the battery following the user requirements and taking into account when it is going to be used again. This behavior complicates the decisions that the algorithm has to take in order to achieve its objectives. But it also gives the opportunity to a new field of research to reduce the variabilities of the aggregated consumption in a concrete area.

- *Complete neural system.* The consumption pattern generated by the algorithm was weighted by the β coefficient. This method serves to establish different priorities for the objectives of the algorithm and achieve good results with the integration of both objectives. However, the decisions were made using non neural elements. In order to get a better performance, a good way to integrate both objectives is the use of a complete neural system to decide the consumption pattern. This modular neural system would respond to different objectives generating only one response in which all the targets are integrated. As one of the major applications of the ANN is the classification and decision making, creating a whole neural system, it is unavoidable to increase the performance of the algorithm. Thus, a first approximation would be using the EDeNC response plus the PV forecast as inputs to an ANN that selects the better objective each time. Adding new objectives (electrical savings) or new elements (storage systems) into the system will consist of adding more inputs to the neural decision-making block, but the training would include all these new targets and elements in order to obtain the required results. Furthermore, for a complete neural system, the PV forecasting can also be done by using an ANN. This predictive module would help to elaborate predictions locally which are essential coordinating the different regions that compose the grid. A disadvantage of getting a large scale neural model is the complexity of the training for each module and the computation capacity of the nodes. But once the right direction of implementing a solution is found, it is the best way to increase the whole performance.

- *More forms of generation.* The most widespread technology used in DG is PV. This is also the form of local generation used for the development of the algorithm proposed in this Thesis. However, there exist other DG technologies among which are multiple renewable energy technologies. One of the most popular forms of DG together with PV is the use of wind power generation. Its integration in the current form of the algorithm is very easy and it can be done in three different ways: substituting the PV, adding a new coefficient to prioritize one local resource or the other, or directly in the generation forecast by adding its response to the one of the PV. Hence, the NGA will continue smoothing the aggregated consumption at the same time that tries to maximize the self-consumption of one or more forms of local generation. In spite of the fast integration in the system, wind power presents a high variability of the resources since its availability is not controllable.
  In addition, getting closer to a real environment, the virtual grid could incorporate different forms of generation typical from the actual grid, such as nuclear power plants or hydraulic ones in order to correlate the matching of the consumption and the generation. By adding different forms of generation, the correlation of the decisions made by the algorithm and the capacity of the system

to respond to them will also be tested. The complexity of the environment offers an opportunity to study the different effects that can happen with the application of this type of algorithms.

- *Spatial simulation grid configuration.* During this Thesis, the aggregated consumption is done instantaneously without taking into account the places where the electricity generated comes from or the different sites where the consumption occurs. As no spatial considerations were taken, the losses around the transportation of electricity could not be taken into account or estimated in the aggregation of the consumption. Therefore, quantification of savings and losses introduced with the NGA could serve to its final deployment in the current grids. This study would also serve as a measurement to improve the conflicting points of the algorithm in the interaction of the facility with the grid. In addition, the inclusion of the spatial dimension to the problem would serve to understand better the effects of the DG and quantify the benefits of using it with respect to the generated electricity coming from the grid. Some special configurations could also be simulated like the island operation of a group of facilities or some microgrids with shared resources. Finally, this study could serve to design and sizing the equipment needed in the grid for the application of the NGA to its future deployment in the SG.

These are some improvements that could be implemented for a fully operational algorithm to be used inside the SG. However, at the same time there exist other applications in which the developments achieved in this Thesis could also be used. Some of these applications are also related with the energy field whereas other applications are related with the signal processing capabilities of the algorithm. These applications are as follows:

- *Microgrid management*[1]. During the development of the evolved neural controllers, the adaptation capabilities of the controllers were tested in a reduced environment in which new users have to adapt their consumption profile to the one of the environment. The use of this property of the EDeNC is very convenient to manage the different consumptions inside micrograms for the optimization of the resources. The idea is that through the self-organization capabilities of the EDeNC, the users are able to optimize the use of the local resources minimizing the interactions with the grid. There exist some limitations inside the microgrid that the EDeNC could use in order to share all the resources among the different users. Moreover, the EDeNC could be used to adapt the consumption of some users to other ones that have priority over the rest, so no energy resource is wasted. In addition, not only the consumption inside the microgrid can be optimized with the EDeNC, the combination of different microgrids could also be coordinated with the use of the NGA.

- *Load shedding.* With the growth of the EV in the transportation sector, a new concern has arisen with respect to the infrastructures to charge its battery. Imagine that inside a garage of a block of flats, all the neighbors have an EV. The parking spot of each neighbor will be equipped with a charger for its vehicle but the common installation of the garage will probably not be changed since the construction of the building. In the case that all the cars need to be charged at the same time, a peak of current is demanded that would soon overload the capacity of the infrastructure. For this reason, different approaches are used to shed the load of the different EVs of the garage. However, this will disconnect those loads that are not considered primary in the system, so some cars will not be charged while the power demanded is reduced. It is in this scenario that the use of the EDeNC could be beneficial in order to regulate the current used to charge the batteries of the cars. Using the EDeNCs, all the EVs could be charging but not at the maximum current of the installation since it will

[1]A microgrid is a localized grouping of electricity generation, energy storage, and loads that could operate connected to a traditional centralized grid (macrogrid)

be shared between all the components. Thus, none of them will be shed and it is assured that they will be charged eventually. In addition, depending on the preferences of the user, different priorities to charge the EVs could also be established while also guaranteeing that all of them are being charged and the installation is not overloaded.

- *Optimize with respect to the price signal.* The NGA was used to optimize the performance of a grid with a global objective of smoothing the aggregated consumption, and a local objective of maximizing the local use of the electricity generation. However, there is more than one objective that can be used in order to schedule the consumption of the different users inside the grid. Following the increase of the electricity bill experienced by users worldwide in the last years, users are concerned about the waste of energy and the electricity prices. Furthermore, the new time pricing would make the users altering their consumption habits in order to achieve higher benefits. This time pricing signal is correlated somehow with the form of the aggregated consumption since for peak periods, the electricity is more expensive than during valley periods. This information can be used to smooth the aggregated consumption while users benefit economically from the savings of consuming during off peaks periods. Thus, the modification of the NGA would consist of using the price signal rather than the normalized aggregated consumption of the grid. This information is more feasible to achieve since the smart meters deployed in the grid infrastructure are prepared for pricing the electricity in real time which will report direct benefits over the users by giving it directly to them.

- *Audio filtering.* Up until now all the applications described were related to the energy field. Another application in the signal processing field for the algorithms developed in this Thesis is in the filtering of audio signals. The first part of the EDeNC was based on the creation of a destructive interference to generate the antiphase signal of an input one. This concept was based on the Active Noise Control (ANC) in which the noise is cancelled by the interference of other signals. The idea is to use the EDeNC not only to cancel a specific pattern but also to coordinate different controllers in order to be able to filter different noises whose origin is unknown and are present in the audio signal. In addition, the EDeNC could also be used to equalize and color the audio signal to those frequencies that the user wants to increase their power and cancel those that are interfering with the rest. The advantage of using the EDeNCs is the adaptability and forecasting properties that they present since they are going to evolve dynamically its response to cancel the different interferences in favor of those that the users want to reinforce. For example, the headphones are using this type of systems to isolate the user, but a good test would be their inclusion in smartphones to cancel the outside noise when speaking or in a theatre to avoid those harmful interferences.

- *Image filtering.* Another application related with the signal processing is the use of the algorithm in another filtering process, in this case one related with the imaging. The EDeNC were evolved using a continuous and derivative of class C^1 signal, and react badly to discontinuities. However, this behavior can be exploited to detect the borders of an image so that it serves to identify the different forms that are inside it. Then, different EDeNC could be coordinated to recognize different forms and classify them. Thus, it could be applied in different tasks related to imaging because they are general purpose algorithms. In addition, the EDeNC can also be used to filter noise of images and equalize them since its continuous and interference creation properties. For example, they could be used to eliminate the salt and pepper noise of black and white images since those perturbations are localized and with the equalization done by EDeNC this noise is easy to be eliminated. Furthermore, other image characteristics can be also altered with the use of these algorithms such as the contrast or brightness of various parts of the image.

- *Pattern recognition.* Last but not least, the EDeNC is based on techniques of the BSS, so it could be used as a pattern recognition algorithm to separate different types of signals. As the EDeNC divides a signal into different pieces part of a whole, it is interesting its application in those environments in which there is no information about the elements that it is composed with and the algorithm could give a possible solution. For example, it can be used in a receptor to identify the different sources of information inside a mixed environment. It can also serve to reconstruct part of the information as some of it is proportioned to the algorithm. So, the EDeNC has to adapt its behavior in order to respect this information and the rest of the signal would be regenerated as one of the EDeNC. For example, this can be applied in a shared environment with multiple messages where some information is known and it is needed to be regenerated at the end of channel.

These applications are only a sample of the multiple ones in which these algorithms can be used. As they are general purpose algorithms with a high component of signal processing, they can be applied to any application related with coordination of ensembles to develop a self-organized task.

7.3 Review of Contributions

This dissertation describes original research carried out by the author. It has not been previously submitted to the Universidad Politécnica de Madrid nor to any other university for the awarding of any degree. Nevertheless, some studies have been done to study different aspects of the energy management, ANNs, cooperative systems and neuroscience. In this Section, all papers published or accepted for publication by the author (11 journals and 6 conferences), together with a number of co-workers, are explained and linked with this Thesis. The corresponding publications are detailed in the following.

The early works of the author were centered in the studies of DSM with local DER in the residential sector in order to maximize the self-consumption. These works where done inside the framework of a national research project called GeDELOS-FV[2]. Three conferences papers show the results achieved in this project. In Castillo-Cagigal et al. (2010b), the results of the self-consumption studies based on the use of the PV generation together with ADSM were published. Then, in Caamaño-Martín et al. (2010), the implementation of the GeDELOS-FV system was introduced. And finally, a paper about the design of battery controller to optimize the self-consumption was published in Castillo-Cagigal et al. (2010a).

Furthermore, the achievements during this research project also allowed the publication of some journal papers. In Castillo-Cagigal et al. (2011a), an ADSM system in which the local energy resources are used to enhance the consumption of a residential user is presented. This paper addresses the improvements caused by the combination of ADSM and storages systems. In Castillo-Cagigal et al. (2011c), the sensor network used in the GeDELOS-FV project was introduced. This platform consists of different embedded systems and smart meters, used to monitor and measure the different variables analyzed during the project.

Following the results achieved during the GeDELOS-FV project, different studies followed around the ideas of developing the concepts of the self-consumption and the ADSM. In Matallanas et al. (2011), a study of the possibilities of self-consumption in residential PV systems in the Spanish grid was carried out. This paper sowed the seeds of this Thesis since Matallanas et al. (2011) studied the different PV systems configurations, the possibilities of enhancing the self-consumption with ADSM and storage systems and an economic analysis of the self-consumption in small grids. It was based on the energy variables obtained from the experiments performed in

[2] GeDELOS-FV: *Gestión de la demanda eléctrica doméstica con tecnología solar fotovoltaisupported by the Plan Nacional de Investigación Científica, Desarrollo e Innovación Tecnológica, 2007-2010, (ENE-2007-66135/ALT).*

GeDELOS-FV. After this study of the self-consumption, a new approach of the ADSM and the maximization of the self-consumption was published in Matallanas et al. (2012). In this paper, the ADSM was addressed from a neural network approach searching for a higher automation of the process. Finally, the last results of the GeDELOS-FV project were published in Masa-Bote et al. (2014). In this last paper, PV forecasts are used to a better integration of the PV systems in the grid plus helping the ADSM to increase the self-consumption.

The studies on the field ADSM and management of the local resources continue, but moving to a new sector and studying a new user coming from an office building with different PV technologies (CPV[3]). With this aim, the SIGMAPLANTAS[4] project was created. A new ADSM controller was developed in this project to meet the energy needs of these users at the same time that the local resources were enhanced. This user counts with the typical consumptions of an office plus a CPV system with an storage system. During the development of the project two more publications were done. In Trujillo et al. (2012), an introduction to the project and the different elements that the installation possesses are described. In Trujillo et al. (2013), the first results of the project are published following the objective of maximizing the self-consumption of the local resources with the ADSM controller developed.

After all the studies made from the local perspective, the author moved to a grid perspective through its collaboration in some research projects. The main relevant project of this topic was the competition *Solar Decathlon Europe*, in which different universities around the world design, build and operate an energetically self-sufficient house connected to the grid in order to compete with each other in different contest related with the house built. The author was part of the monitoring team in charge of designing, implementing and installing the monitoring system for the different houses of the competition. The competition was held in September 2012 and three papers were published with the lessons learned in this competition. In Navarro et al. (2014), a general overview of the competition is presented in which all its parts are described and the results obtained are analyzed. In Matallanas et al. (2014), the electrical contest is analyzed and the different results of the participants teams were shown. Finally, in Rodriguez-Ubiñas et al. (2014), the paper addresses the use of passive techniques, broadly architectonics techniques, to reduce the energy consumption in the electrical grid.

During the development of this Thesis, the author did a research intern in the Centre for Theoretical Neuroscience. The author wanted to increase his knowledge in the biological part of the ANNs and also wanted to understand better the construction of new models of neurons that resemble the biological ones. During his internship, the work done was published in a conference and in a journal paper. In Rezai et al. (2014a) and Rezai et al. (2014b), two parts of the brain were studied: the Anterior IntraParietal area (AIP) and Caudal IntraParietal area (CIP) of the brain of monkeys. The idea was to build a computational model of these two parts of the brain related with the visual guided grasping function in order to obtain the parametrization necessary to implement in a robotic system.

Finally, the author has been also developing some studies in the coordination and cooperation of individual ensembles. This work motivated the coordination of the different neural ensembles of the EDeNC during the development of this Thesis. Following this research line, two more journal papers were published. In Castillo-Cagigal et al. (2014), a division of labor model as a discrete-time dynamical system is defined in which their different properties are studied and an algorithm to modify its dynamical behavior was suggested. In Castillo-Cagigal et al. (2016), a multi-agent periodic environment is coordinated by an algorithm that varies the internal frequency of each individual in order to obtain the results pursued.

[3]Concentrated PV (CPV) technology uses optics such as lenses or curved mirrors to concentrate a large amount of sunlight onto a small area of solar PV cells to generate electricity.

[4] SIGMAPLANTAS: La innovación en las plantas y modelos de sistemas de Concentración Fotovoltaica en España. Ministerio de Ciencia e Innovación, programa INNPACTO (CIN/699/2011) IPT-2011-1468-920000.

Bibliography

Abdelaziz, E. A., Saidur, R., and Mekhilef, S. (2011). A review on energy saving strategies in industrial sector. *Renewable and Sustainable Energy Reviews*, 15(1):150–168.

Achanta, R. (2012). Long term electric load forecasting using neural networks and support vector machines. *International Journal of Computer Science And Technology*, 3(1):266–269.

Albadi, M. and El-Saadany, E. (2007). Demand response in electricity markets: An overview. In *Power Engineering Society General Meeting, 2007. IEEE*, pages 1–5, Piscataway, NJ. IEEE.

Amari, S. (1972). Learning patterns and pattern sequences by self-organizing nets of threshold elements. *Computers, IEEE Transactions on*, C-21(11):1197–1206.

Arbib, M. A. (2003). *The handbook of brain theory and neural networks.* The MIT Press, Cambridge, Mass.

Barker, P. and de Mello, R. (2000). Determining the impact of distributed generation on power systems. i. radial distribution systems. In *Power Engineering Society Summer Meeting, 2000. IEEE*, volume 3, pages 1645–1656, Piscataway, NJ. IEEE.

Basheer, I. and Hajmeer, M. (2000). Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods*, 43(1):3–31.

Beer, R. D. (1995). On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior*, 3(4):469–509.

Beer, R. D. (1997). The dynamics of adaptive behavior: A research program. *Robotics and Autonomous Systems*, 20(2—4):257 – 289.

Bellman, R. E. (1957). *Dynamic Programming.* Princeton University Press, Princeton.

Bengio, Y. (2009). Learning deep architectures for ai. *Foundations and Trends in Machine Learning*, 2(1):1–127.

Benjamin, B., Gao, P., McQuinn, E., Choudhary, S., Chandrasekaran, A., Bussat, J.-M., Alvarez-Icaza, R., Arthur, J., Merolla, P., and Boahen, K. (2014). Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations. *Proceedings of the IEEE*, 102(5):699–716.

Berlinski, D. (2000). *The Advent Of The Algorithm.* Harcourt, San Diego.

Bishop, C. M. (2006). *Pattern recognition and machine learning.* Springer, New York.

Blum, A. L. and Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1-2):245 – 271.

Blume, S. W. (2007). *Electric power system basics: For the nonelectrical professional.* Wiley-Interscience, Hoboken, N.J.

Boyle, G. (2007). *Renewable electricity and the grid: the challenge of variability.* Earthscan, London.

Brain, M. and Roos, D. (2000). How power grids work. http://science.howstuffworks.com/environmental/energy/power.htm. [Online; accessed 29 June 2015].

Braitenberg, V. (1986). Two views of the cerebral cortex. In Palm, G. and Aertsen, A., editors, *Brain Theory*, pages 81–96. Springer-Verlag, Berlin.

Buhman, M. D. (1998). *Combining Artificial Neural Nets: Ensemble and Modular Multi-Net Systems.* Springer-Verlag, London, NY.

Buhman, M. D. (2003). *Radial Basis Funtions: Theory and Implementation.* Cambridge University Press, Cambridge.

Bunnoon, P. (2011). Mid-term load forecasting based on neural network algorithm: a comparison of models. *International Journal of Computer and Electrical Engineering*, 3(4):600–605.

Caamaño-Martín, E., Castillo-Cagigal, M., Masa-Bote, D., Álvaro Gutiérrez, Matallanas, E., Monasterio-Huelin, F., and Jiménez-Leube, J. (2010). Autoconsumo de energía solar fotovoltaica con gestión activa de la demanda: El sistema gedelos-fv. In *Actas del IV Conferencia Latinoamericana de Energía Solar (IV ISES CLA) y XVII Simposio Peruano de Energía Solar (XVII- SPES)*, Cuzco, Perú. Asociacion Peruana de Energia Solar y del Ambiente.

Canadian Electricity Association (2010). *The smart grid: a pragmatic approach.* Canadian Electronic Library, Ottawa, Ontario.

Castillo-Cagigal, M. (2014). *A swarm intelligence approach based on coupled oscillators: an application in demand side management with photovoltaic distributed generation.* PhD thesis, Universidad Politécnica de Madrid.

Castillo-Cagigal, M., Caamaño-Martín, E., Matallanas, E., Masa-Bote, D., Gutiérrez, A., Monasterio-Huelin, F., and Jiménez-Leube, J. (2011a). Pv self-consumption optimization with storage and active dsm for the residential sector. *Solar Energy*, 85(9):2338–2348.

Castillo-Cagigal, M., Gutiérrez, A., Monasterio-Huelin, F., Caamaño-Martín, E., Masa-Bote, D., and Jiménez-Leube, J. (2011b). A semi-distributed electric demand-side management system with PV generation for self-consumption enhancement. *Energy Conversion and Management*, 52(7):2659–2666.

Castillo-Cagigal, M., Matallanas, E., Gutiérrez, A., Monasterio-Huelin, F., Caamaño-Martín, E., Masa-Bote, D., and Jiménez-Leube, J. (2011c). Heterogeneous collaborative sensor network for electrical management of an automated house with pv energy. *Sensors*, 11(12):11544–11559.

Castillo-Cagigal, M., Matallanas, E., Masa-Bote, D., Caamaño-Martín, E., Gutiérrez, A., Monasterio-Huelin, F., and Jiménez-Leube, J. (2010a). Self-consumption enhancement with storage system and demand-side management: Gedelos-pv system. In *5th International Renewable Energy Storage Conference IRES 2010*, pages 1–5, Berlin, Germany. EUROSOLAR.

Castillo-Cagigal, M., Matallanas, E., Monasterio-Huelin, F., Caamano-Martin, E., and Gutierrez, A. (2016). Multi-frequency coupled oscillators for distributed multi-agent coordination. *IEEE Transactions on Industrial Informatics*, 12(3):941–951.

Castillo-Cagigal, M., Matallanas, E., Navarro, I., Caamano-Martin, E., Monasterio-Huelin, F., and Gutierrez, A. (2014). Variable threshold algorithm for division of labor analyzed as a dynamical system. *Cybernetics, IEEE Transactions on*, 44(12):2242–2252.

Castillo-Cagigal, M., no Martín, E. C., Gutiérrez, A., Masa-Bote, D., Monasterio, F., Porro, J., Matallanas, E., and Leube, J.-J. (2010b). Self-consumption of pv electricity with active demand side management: the gedelos-pv system. In *Proceedings of the 25th European Photovoltaic Solar Energy Conference*, pages 4866–4870, Munich, Germany. WIP-Renewable Energies.

Chawla, G., Sachdev, M., and Ramakrishna, G. (2005). Artificial neural network applications for power system protection. In *Electrical and Computer Engineering, 2005. Canadian Conference on*, pages 1954–1957, Piscataway, NJ. IEEE.

Chay, T. R. (1981). A model for biological oscillations. *Proceedings of the National Academy of Sciences*, 78(4):2204–2207.

Chen, H., Cong, T. N., Yang, W., Tan, C., Li, Y., and Ding, Y. (2009). Progress in electrical energy storage system: A critical review. *Progress in Natural Science*, 19(3):291 – 312.

Cohen, M. A. and Grossberg, S. (1983). Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. *Systems, Man and Cybernetics, IEEE Transactions on*, 13(5):815–826.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Math. Control Signals Systems*, 2(4):303–314.

Daneshi, H., Shahidehpour, M., and Choobbari, A. (2008). Long-term load forecasting in electricity market. In *Electro/Information Technology, 2008. EIT 2008. IEEE International Conference on*, pages 395–400, Piscataway, NJ. IEEE.

Dauce, E., Quoy, M., Cessac, B., Doyon, B., and Samuelides, M. (1998). Self-organization and dynamics reduction in recurrent networks: stimulus presentation and learning. *Neural Networks*, 11(3):521 – 533.

Dayan, P. and Abbott, L. F. (2001). *Theoretical neuroscience: computational and mathematical modeling of neural systems*. The MIT Press, Cambridge, Mass.

Deilami, S., Masoum, A., Moses, P., and Masoum, M. (2011). Real-time coordination of plug-in electric vehicle charging in smart grids to minimize power losses and improve voltage profile. *Smart Grid, IEEE Transactions on*, 2(3):456–467.

Draye, J.-P., Pavisic, D., Cheron, G., and Libert, G. (1995). Adaptative time constants improve the prediction capability of recurrent neural networks. *Neural Processing Letters*, 2(3):12–16.

Ebron, S., Lubkeman, D., and White, M. (1990). A neural network approach to the detection of incipient faults on power distribution feeders. *Power Delivery, IEEE Transactions on*, 5(2):905–914.

Edenhofer, O., Pichs-Madruga, R., Sokona, Y., Seyboth, K., Matschoss, P., Kadner, S., Zwickel, T., Eickemeier, P., Hansen, G., Schlömer, S., and von Stechow, C. (2011). Special report on renewable energy sources and climate change mitigation. Technical Report 1472, IPCC.

Ekanayake, J., Jenkins, N., Liyanage, K., Wu, J., and Yokoyama, A. (2012). *Smart Grid: Technology and Applications*. John Wiley & Sons, Inc., Hoboken, N.J.

El-Fergany, A., Yousef, M., and El-Alaily, A. (2001). Fault diagnosis in power systems-substation level-through hybrid artificial neural networks and expert system. In *Transmission and Distribution Conference and Exposition, 2001 IEEE/PES*, volume 1, pages 207–211, Piscataway, NJ. IEEE.

El-hawary, M. E. (2014). The smart grid—state-of-the-art and future trends. *Electric Power Components and Systems*, 42(4):239–250.

Eliasmith, C. and Anderson, C. H. (2003). *Neural engineering : computation, representation, and dynamics in neurobiological systems.* The MIT Press, Cambridge, Mass.

Eliasmith, C., Stewart, T. C., Choo, X., Bekolay, T., DeWolf, T., Tang, Y., and Rasmussen, D. (2012). A large-scale model of the functioning brain. *Science*, 338(6111):1202–1205.

Ellabban, O., Abu-Rub, H., and Blaabjerg, F. (2014). Renewable energy resources: Current status, future prospects and their enabling technology. *Renewable and Sustainable Energy Reviews*, 39:748 – 764.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14(2):179–211.

European Commission (2006). *European SmartGrids Technology Platform: Vision and strategy for Europe's electricity networks of the future.* EUR-OP, Luxembourg.

European Commission (2009). *ICT for a Low Carbon Economy. Smart Electricity Distribution Network.* European Commission Information Society and Media, Brussels, Belgium.

Fang, X., Misra, S., Xue, G., and Yang, D. (2012). Smart grid - the new and improved power grid: A survey. *IEEE Communications Surveys and Tutorials*, 14(4):944–980.

Farhangi, H. (2010). The path of the smart grid. *Power and Energy Magazine, IEEE*, 8(1):18–28.

Feilat, E. and Bouzguenda, M. (2011). Medium-term load forecasting using neural network approach. In *Innovative Smart Grid Technologies - Middle East (ISGT Middle East), 2011 IEEE PES Conference on*, pages 1–5, Piscataway, NJ. IEEE.

Fox-Penner, P. (2010). *Smart Power: Climate Change, the Smart Grid, and the Future of Electric Utilities.* Island Press, Washington, D.C.

Freeman, W. J. (1995). *Societies of brains.* Lawrence Erlbaum, Hillsdale, NJ.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202.

Gelazanskas, L. and Gamage, K. A. (2014). Demand side management in smart grid: A review and proposals for future direction. *Sustainable Cities and Society*, 11:22 – 30.

Gellings, C. (1985). The concept of demand-side management for electric utilities. *Proceedings of the IEEE*, 73(10):1468–1470.

Germond, A., Macabrey, N., and Baumann, T. (1993). Application of artificial neural networks to load forecasting. In Sobajic, D. J., editor, *Neural Network Computing For the Electric Power Industry: Proceedings Of The 1992 INNS Summer Workshop*, pages 448–453, Hillsdale, NJ. Lawrence Erlbaum Associates.

Gharavi, H. and Ghafurian, R. (2011). Smart grid: The electric energy system of the future. *Proceedings of the IEEE*, 99(6):917–921.

Glover, F. and Laguna, M. (1999). Tabu search. In *Handbook of combinatorial optimization*, pages 2093–2229. Springer, New York City, USA.

Golberg, D. E. (2006). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley, Indiana, USA.

Goldbeter, A. (1997). *Biochemical Oscillations and Cellular Rhythms: The Molecular Bases of Periodic and Chaotic Behaviour*. Cambridge University Press, Cambridge, UK.

Goser, K. F. (1996). Implementation of artificial neural networks into hardware: Concepts and limitations. *Mathematics and Computers in Simulation*, 41(1–2):161–171.

Grossberg, S. (1969). Embedding fields: A theory of learning with physiological implications. *Journal of Mathematical Psychology*, 6(2):209–239.

Grossberg, S. (1974). Classical and instrumental learning by neural networks. In Rosen, R. J. and Snell, F. M., editors, *Progress in theoretical biology*, pages 51–141. Academic Press, New York.

Grossberg, S. (1987). Competitive learning: From interactive activation to adaptive resonance. *Cognitive Science*, 11(1):23 – 63.

Guckenheimer, J. and Holmes, P. (1983). *Nonlinear oscillations, dynamical systems, and bifurcations of vector fields*. Springer-Verlag, New York.

Haidar, A. M. A., Mustafa, M. W., Ibrahimc, F. A. F., and Ahmedd, I. A. (2011). Transient stability evaluation of electrical power system using generalized regression neural networks. *Applied Soft Computing*, 11(4):3558–3570.

Haque, M. T. and Kashtiban, A. (2007). Application of neural networks in power systems: A review. *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, 1(6):889–893.

Hayashi, Y. and Iwamoto, S. (1994). Long-term load forecasting using improved recurrent neural network. *Electrical Engineering in Japan*, 114(8):41–54.

Haykin, S. (2009). *Neural networks and learning machines*. Prentice Hall/Pearson, New York.

Haykin, S. S. (2000). *Unsupervised adaptive filtering: Volume I, Blind source separation*. Wiley, New York, NY.

Hebb, D. O. (1949). *The organization of behavior: a neuropsychological theory*. Wiley, New York.

Herculano-Houzel, S. (2009). The human brain in numbers: a linearly scaled-up primate brain. *Frontiers in Human Neuroscience*, 3(31):1–11.

Hilbert, M. and López, P. (2011). The world's technological capacity to store, communicate, and compute information. *Science*, 332(6025):60–65.

Hinton, G. E. and Sejnowski, T. J. (1983). Optimal perceptual inference. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 448–453, Piscataway, NJ. IEEE.

Hippert, H., Pedreira, C., and Souza, R. (2001). Neural networks for short-term load forecasting: a review and evaluation. *Power Systems, IEEE Transactions on*, 16(1):44–55.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems.* University of Michigan Press, Ann Arbor, MI.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558.

IEA (2011). *Technology Roadmap: Smart Grids.* International Energy Agency, Paris.

IEA (2013). *Global EV outlook.* International Energy Agency, Paris, France.

IEA (2014a). *Energy Technology Perspectives.* International Energy Agency, Paris.

IEA (2014b). *Technology Roadmap: Solar Photovoltaic Energy.* International Energy Agency, Paris, France.

IEA (2014c). *Tracking clean energy progress.* International Energy Agency, Paris.

IEA (2014d). *World energy outlook 2014.* International Energy Agency, Paris, France.

IEC (2011). Electrical Energy Storage: White Paper. Technical Report IEC WP EES:2011-12(en), International Electrotechnical Commission.

II, R. C. G., Wang, L., and Alam, M. (2011). The impact of plug-in hybrid electric vehicles on distribution networks: A review and outlook. *Renewable and Sustainable Energy Reviews*, 15(1):544 – 553.

Imen, L., Mouhamed, B., and Djamel, L. (2013). Economic dispatch using classical methods and neural networks. In *Electrical and Electronics Engineering (ELECO), 2013 8th International Conference on*, pages 172–176, Piscataway, NJ. IEEE.

Izhikevich, E. M. (2007). *Dynamical systems in neuroscience : the geometry of excitability and bursting.* The MIT Press, Cambridge, Mass.

Jaeger, H. (2001). The "echo state" approach to analysing and training recurrent neural networks. Technical Report GMD Report 148, German National Research Institute for Computer Science.

Jain, A., Mao, J., and Mohiuddin, K. (1996). Artificial neural networks: a tutorial. *Computer*, 29(3):31–44.

Jordan, M. L. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. In *Proceedings of the 8th annual conference of the cognitive science society*, Hillsdale, NJ. Lawrence Erlbaum Associates.

Kandel, E. R., Schwartz, J. H., Jessell, T. M., Siegelbaum, S. A., and Hudspeth, A. J. (2013). *Principles of Neural Science.* McGraw-Hill, New York.

Karami, A. (2011). Power system transient stability margin estimation using neural networks. *Electrical Power and Energy Systems*, 33(4):983–991.

Kim, J. H. and Shcherbakova, A. (2011). Common failures of demand response. *Energy*, 36(2):873–880.

Kirkpatrick, M. (1982). Sexual selection and the evolution of female choice. *Evolution*, 36(1):1–12.

Kirtley, J. L. (2010). *Electric Power Principles: Sources, Conversion, Distribution and Use.* John Wiley & Sons, Chichester, UK.

Kohavi, R. and Provost, F. (1998). Glossary of terms. *Machine Learning*, 30(2-3):271–274.

Kohonen, T. (1977). *Associative Memory: A System-Theoretical Approach*. Springer-Verlag, Berlin.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C., Bottou, L., and Weinberger, K., editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., New York.

Kundur, P., Balu, N. J., and Lauby, M. G. (1994). *Power system stability and control*. McGraw-Hill, New York.

Kuznetsov, Y. A. (1998). *Elements of applied bifurcation theory*. Springer-Verlag, New York.

Le, Q. V., Monga, R., Devin, M., Chen, K., Corrado, G. S., Dean, J., and Ng, A. Y. (2012). Building high-level features using large scale unsupervised learning. In *International Conference on Machine Learning*, pages 81–88, Madison, WI. Omnipress.

Lopes, J. P., Hatziargyriou, N., Mutale, J., Djapic, P., and Jenkins, N. (2007). Integrating distributed generation into electric power systems: A review of drivers, challenges and opportunities. *Electric Power Systems Research*, 77(9):1189 – 1203.

Luthra, S., Kumar, S., Kharb, R., Ansari, M. F., and Shimmi, S. (2014). Adoption of smart grid technologies: An analysis of interactions among barriers. *Renewable and Sustainable Energy Reviews*, 33:554–565.

Lyapunov, A. M. (1992). The general problem of the stability of motion. *International Journal of Control*, 55(3):531–534.

Mandic, D. P. and Chambers, J. A. (2001). *Recurrent neural networks for prediction: Learning algorithms, architectures, and stability*. John Wiley, Chichester.

Marsland, S. (2009). *Machine learning: an algorithmic perspective*. CRC Press, Boca Raton.

Masa-Bote, D., Castillo-Cagigal, M., Matallanas, E., Caamaño-Martín, E., Gutiérrez, A., Monasterio-Huelin, F., and Jiménez-Leube, J. (2014). Improving photovoltaics grid integration through short time forecasting and self-consumption. *Applied Energy*, 125(0):103–113.

Matallanas, E., Castillo-Cagigal, M., Caamaño-Martín, E., Masa-Bote, D., Gutiérrez, A., and Monasterio-Huelin, F. (2011). Analysis of the self-consumption possibilities in small grid-connected photovoltaic systems in spain. In *Proceedings of the 26th European Photovoltaic Solar Energy Conference*, pages 4619–4624, Munich, Germany. WIP-Renewable Energies.

Matallanas, E., Castillo-Cagigal, M., Gutiérrez, A., Monasterio-Huelin, F., Caamaño-Martín, E., Masa, D., and Jiménez-Leube, J. (2012). Neural network controller for Active Demand-Side Management with PV energy in the residential sector. *Applied Energy*, 91(1):90–97.

Matallanas, E., Solórzano, J., Castillo-Cagigal, M., naki Navarro, I., no Martín, E. C., Egido, M. A., and Gutiérrez, A. (2014). Electrical energy balance contest in solar decathlon europe 2012. *Energy and Buildings*, 83(0):36–43.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.

Mecrow, B. C. and Jack, A. G. (2008). Efficiency trends in electric machines and drives. *Energy Policy*, 36(12):4336–4341.

Mets, K., Verschueren, T., Haerick, W., Develder, C., and De Turck, F. (2010). Optimizing smart energy control strategies for plug-in hybrid electric vehicle charging. In *Network Operations and Management Symposium Workshops (NOMS Wksps), 2010 IEEE/IFIP*, pages 293–299, Piscataway, NJ. IEEE.

Minsky, M. and Papert, S. A. (1969). *Perceptrons*. The MIT Press, Cambridge, Mass.

Misra, J. and Saha, I. (2010). Artificial neural networks in hardware: A survey of two decades of progress. *Neurocomputing*, 74(1–3):239 – 255.

MIT (2011). *The Future of the Electric Grid*. Massachusetts Institute of Technology, Boston, USA.

Mohd, A., Ortjohann, E., Schmelter, A., Hamsic, N., and Morton, D. (2008). Challenges in integrating distributed energy storage systems into future smart grid. In *Industrial Electronics, 2008. ISIE 2008. IEEE International Symposium on*, pages 1627–1632, Piscataway, NJ. IEEE.

Molburg, J. C., Kavicky, J. A., and Picel, K. C. (2007). The design, construction and operation of long-distances high-voltage electricity transmission technologies. Technical Report AN/EVS/TM/08-4, Argonne National Laboratory.

Momoh, J. (2012). *Smart Grid: Fundamentals of Design and Analysis*. IEEE Press, Piscataway, N.J.

Morabito, V. (2015). *Big Data and Analytics: Strategic and Organizational Impacts*. Springer International Publishing, Cham.

Navarro, I., Gutiérrez, A., Montero, C., Rodríguez-Ubiñas, E., Matallanas, E., Castillo-Cagigal, M., Porteros, M., Solórzano, J., Caamaño-Martín, E., Egido, M. A., Páez, J. M., and Vega, S. (2014). Experiences and methodology in a multidisciplinary energy and architecture competition: Solar decathlon europe 2012. *Energy and Buildings*, 83(0):3–9.

Newsham, G. R. and Bowker, B. G. (2010). The effect of utility time-varying pricing and load control strategies on residential summer peak electricity use: A review. *Energy Policy*, 38(7):3289–3296.

Ostojic, D. and Heydt, G. (1991). Transient stability assessment by pattern recognition in the frequency domain. *Power Systems, IEEE Transactions on*, 6(1):231–237.

Palensky, P. and Dietrich, D. (2011). Demand side management: Demand response, intelligent energy systems, and smart loads. *Industrial Informatics, IEEE Transactions on*, 7(3):381–388.

Panajotovic, B., Jankovic, M., and Odadzic, B. (2011). Ict and smart grid. In *Telecommunication in Modern Satellite Cable and Broadcasting Services (TELSIKS), 2011 10th International Conference on*, pages 118–121, Piscataway, NJ. IEEE.

Papagiannis, G., Dagoumas, A., Lettas, N., and Dokopoulos, P. (2008). Economic and environmental impacts from the implementation of an intelligent demand side management system at the european level. *Energy Policy*, 36(1):163–180.

Park, D., El-Sharkawi, M., Marks, R.J., I., Atlas, L., and Damborg, M. (1991). Electric load forecasting using an artificial neural network. *Power Systems, IEEE Transactions on*, 6(2):442–449.

Park, J. H., Kim, Y. S., Eom, I. K., and Lee, K. Y. (1993). Economic load dispatch for piecewise quadratic cost function using hopfield neural networks. *Power Systems, IEEE Transactions on*, 8(3):1030–1038.

Pasemann, F. (1993). Discrete dynamics of two neuron networks. *Open Systems & Information Dynamics*, 2(1):49–66.

Pasemann, F., Hild, M., and Zahedi, K. (2003). SO(2)-Networks as Neural Oscillators. In Mira, J. and Alvarez, J. R., editors, *Computational Methods in Neural Modeling, Proceedings IWANN 2003, LNCS 2686*, Berlin, Germany. Springer.

Pearlmutter, B. (1995). Gradient calculations for dynamic recurrent neural networks: a survey. *Neural Networks, IEEE Transactions on*, 6(5):1212–1228.

Pearlmutter, B. A. (1989). Learning state space trajectories in recurrent neural networks. *Neural Computation*, 1(2):263–269.

Pearlmutter, B. A. (1990). Dynamic recurrent neural networks. Technical Report CMU-CS-90-196, School of Computer Science Carnegie Mellon University.

Pepermans, G., Driesen, J., Haeseldonckx, D., Belmans, R., and D'haeseleer, W. (2005). Distributed generation: definition, benefits and issues. *Energy Policy*, 33(6):787–798.

Rahimi, F. and Ipakchi, A. (2010). Demand response as a market resource under the smart grid paradigm. *Smart Grid, IEEE Transactions on*, 1(1):82–88.

Ramchurn, S. D., Vytelingum, P., Rogers, A., and Jennings, N. (2011). Agent-based control for decentralised demand side management in the smart grid. In *The 10th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, pages 5–12, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Ramón y Cajal, S. (1909). *Histologie du système nerveux de l'homme & des vertébrés.* A. Maloine, Paris.

REE (1998). *Atlas de la demanda eléctrica Española. Proyecto INDEL.* Red Eléctrica de España. S.A., Madrid.

Rezai, O., Kleinhans, A., Matallanas, E., Selby, B., and Tripp, B. P. (2014a). A computational model of the anterior intraparietal area. In *Proceedings of the 10th Bernstein Conference*, pages 88–89, Goettingen, Germany. BFNT Goettingen.

Rezai, O., Kleinhans, A., Matallanas, E., Selby, B., and Tripp, B. P. (2014b). Modelling the shape hierarchy for visually guided grasping. *Frontiers in Computational Neuroscience*, 8(132).

Ribeiro, P. F., Duque, C. A., da Silveira, P. M., and Cerqueira, A. S. (2014). *Power Systems Signal Processing for Smart Grids.* John Wiley & Sons, Inc., Hoboken, N.J.

Rodriguez-Ubiñas, E., Montero, C., Porteros, M., Vega, S., Navarro, I., Castillo-Cagigal, M., Matallanas, E., and Gutiérrez, A. (2014). Passive design strategies and performance of net energy plus houses. *Energy and Buildings*, 83(0):10–22.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.

Rosenblatt, F. (1961). *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms.* Spartan Books, Washington D.C.

Rumelhart, D. E. and McClelland, J. L. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition.* The MIT Press, Cambridge, Mass.

Samantaray, S. R. (2014). Letter to the editor: Smart grid initiatives in india. *Electric Power Components and Systems*, 42(3-4):262–266.

Schewe, P. F. (2007). *The grid: a journey through the heart of our electrified world*. J. Henry Press, Washington, D.C.

Sen, S., Sengupta, S., and Chakrabarti, A. (2015). *Electricity pricing : regulated, deregulated and smart grid systems*. CRC Press, Boca Raton.

Sobajic, D. and Pao, Y.-H. (1989). Artificial neural-net based dynamic security assessment for electric power systems. *Power Systems, IEEE Transactions on*, 4(1):220–228.

Srilatha, N., Yesuratnam, G., and Deepthi, M. (2014). Rbf neural network approach for security assessment and enhancement. In *T D Conference and Exposition, 2014 IEEE PES*, pages 1–5, Piscataway, NJ. IEEE.

Strbac, G. (2008). Demand side management: Benefits and challenges. *Energy Policy*, 36(12):4419 – 4426.

Strogatz, S. H. (1994). *Nonlinear dynamics and chaos : with applications to physics, biology, chemistry, and engineering*. Westview Press, Cambridge, Massachusetts.

Swarup, K. S. and Corthis, P. B. (2006). Power system static security assessment using self-organizing neural network. *Journal of the Indian Institute of Science*, 86(4):327–342.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 1–9, Piscataway, NJ. IEEE.

Thiaw, L., Sow, G., and Fall, S. (2014). The application of artificial neural networks in the estimation of wind speed: A case study. In *Proceedings of the 2014 First International Conference on Systems Informatics, Modelling and Simulation*, pages 8–13, Washington, DC, USA. IEEE Computer Society.

Torriti, J., Hassan, M. G., and Leach, M. (2010). Demand response experience in Europe: Policies, programmes and implementation. *Energy*, 35(4):1575 – 1583.

Trujillo, P., Alamillo, C., Gil, E., Óscar de la Rubia, Martínez1, M., Rubio, F., Cadavid, A., Navarro, J., Hillenbrand, S., Ballesteros-Sánchez, I., Castillo-Cagigal, M., Masa-Bote, D., Matallanas, E., no Martín, E. C., and Álvaro Gutiérrez (2012). Cpv hybrid system in isfoc building, first results. In *Proceedings of the8th International Conference on Concentrat ing Photovoltaic Systems*, pages 360–363, Maryland, USA. American Institute of Physics.

Trujillo, P., Alamillo, C., Martínez, M., Óscar de la Rubia, Rubio, F., Masa-Bote, D., Castillo-Cagigal, M., Matallanas, E., o-Martín, E. C., and Álvaro Gutiérrez (2013). Instituto de sistemas fotovoltaicos de concentración con centration photovoltaics hybrid system first year of operation and improvements. *Progress in Photovoltaics: Research and Applications*, 21(6):1260–1275.

Tsekouras, G., Hatziargyriou, N., and Dialynas, E. (2006). An optimized adaptive neural network for annual midterm energy forecasting. *Power Systems, IEEE Transactions on*, 21(1):385–391.

Tu, B. P. (2006). Metabolic cycles as an underlying basis of biological oscillations. *Nat Rev Mol Cell Biol*, 7(9):696–701.

UK Department of Energy & Climate Change (2009). *Smarter Grids: The Opportunity*. UK Department of Energy & Climate Change, London.

United States (2007). *Energy independence and security act of 2007*. U.S. G.P.O., Washington, D.C.

U.S. Department of Energy (2009). *Smart Grid System Report*. U.S. Department of Energy, Washington, DC.

von Meier, A. (2006). *Electric power systems: a conceptual introduction*. John Wiley & Sons, Inc., Hoboken, N.J.

von Neumann, J. (1951). *The general and logical theory of automata*, pages 1–41. Wiley, Pasadena CA.

Vu, K., Begouic, M., and Novosel, D. (1997). Grids get smart protection and control. *Computer Applications in Power, IEEE*, 10(4):40–44.

Wade, N., Taylor, P., Lang, P., and Jones, P. (2010). Evaluating the benefits of an electrical energy storage system in a future smart grid. *Energy Policy*, 38(11):7180 – 7188.

Wang, F. (2011). Fault diagnosis for power systems based on neural networks. In *Software Engineering and Service Science (ICSESS), 2011 IEEE 2nd International Conference on*, pages 352–355, Piscataway, NJ. IEEE.

Werbos, P. (1974). *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Science*. PhD thesis, Harvard University.

Widrow, B. and Hoff, M. E. (1960). Adaptive switching circuits. In *IRE WESCON Convention Record*, pages 96–104, New York, N.Y. Institute of Radio Engineers.

Wiener, N. (1948). *Cybernetics or control and communication in the animal and the machine*. The MIT Press, Cambridge, Massachusetts.

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

Williams, R. J. and Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2):270–280.

Wilson, H. R. and Cowan, J. D. (1972). Excitatory and inhibitory interactions in localized populations of model neurons. *Biophysical Journal*, 12(1):1–24.

Wood, G. and Newborough, M. (2003). Dynamic energy-consumption indicators for domestic appliances: environment, behaviour and design. *Energy and Buildings*, 35(8):821–841.

Xu, Z., Xue, Y., and Wong, K. P. (2014). Recent advancements on smart grids in china. *Electric Power Components and Systems*, 42(3-4):251–261.

Yalcinoz, T. and Short, M. (1998). Neural networks approach for solving economic dispatch problem with transmission capacity constraints. *Power Systems, IEEE Transactions on*, 13(2):307–313.

Zervos, A., Lins, C., and Muth, J. (2010). *Re-thinking 2050: A 100% renewable energy vision for the European Union*. European Renewable Energy Council, Brussels, Belgium.

Zurada, J. M. (1992). *Introduction to artificial neural systems*. West, St. Paul.