Doctoral Dissertation
Doctoral Program in Computer and Control Engineering (35^{th}cycle)

Distributed platform for multi-model co-simulations in smart grids

By

Matteo Orlando

******

Supervisor(s):
Prof. Massimo Poncino, Supervisor

Doctoral Examination Committee:
Prof. Olindo Isabella, Referee, University of Delft University of Technology
Prof. Alessandro Margara, Referee, Politecnico di Milano
Prof. Haroon Mahmood, FAST National University of Computer and Emerging Sciences, Islamabad Campus
Prof. Yukai Chen, Interuniversity Microelectronics Centre

Politecnico di Torino
2023

Declaration

I hereby declare that, the contents and organization of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

Matteo Orlando
2023

* This dissertation is presented in partial fulfillment of the requirements for **Ph.D. degree** in the Graduate School of Politecnico di Torino (ScuDo).

*I would like to dedicate this thesis to my loving parents, my sister, my grandmother,
and to all the friends, old and new, that sustained me in this experience*

Acknowledgements

I would like to acknowledge the EDA Group of the Department of Control and Computer Engineering that supported me since I joined the group and their essential helpduring this three-year of Ph.D. activities. In particular, I would like to thank Prof. Massimo Poncino for giving me the opportunity of undertaking my Ph.D. career and Prof. Edoardo Patti, Prof. Sara Vinco, and Prof. Lorenzo Bottaccioli for their guidance that allowed my competencies to grow. Last, but not least, I would like to express my thank to Davide, Evelina, and Claudia that shared the Ph.D. experience with me. I would also like to acknowledge Alessandro that shared with me the joys and the effort for the RURITAGE project.

Abstract

Nowadays legacy electric grids are facing a big challenge: the steady increase in energy demand is increasing the need for a renovation of the systems which requires the introduction of technologies that can facilitate monitoring and increase the automation capabilities. In addition to this, the raising necessity to shift from fossil-fuel is rapidly increasing the diffusion of renewable energy sources. These two challenges force us to rethink the way in which the grid is operated and require us to switch to the Smart Grid paradigm. This paradigm aims to redesign the old grid by introducing Information and Communication Technologies (ICT) that can support the monitoring and the automation mechanism, but also it allows us to take into account the various distributed renewable energy resources that will be more and more diffused. This dissertation addresses these 2 challenges by using the Internet Of Things (IoT) to provide a possible solution to the first challenge, and by studying solutions for the new actors that will enter the energy market with their own distributed renewable energy systems.

Firstly the dissertation proposes an IoT architecture to enable communication among the different entities of the Smart grid to allow both an accurate monitoring of the grid and also to enable remote actuation and therefore increase the automation of the grid. This architecture makes use of well-known ICT tools to assess its capabilities and its resilience. Together with that, the dissertation introduces a design guideline for a 3-phase smart-meter and proposes a related prototype. This prototype realized with low-cost and open-source hardware, exploits the communication infrastructure to self-configure and auto-update itself, but also to achieve a self-healing grid.

Then the dissertation analyzes the characteristics of the new actor that will appear in the smart grid, in particular: i) the prosumers, i.e. those customers that can both consume or self-produce the energy they need, ii) the Energy Aggregator, i.e. a third

party entity that aggregates the energy demand and request of multiple customers, iii) the Renewable Energy Community, i.e. a group of prosumers that join together to share a common renewable energy system which can at least partially satisfy their energy demand. Nowadays the most common technology used by those actors are PhotoVoltaic (PV) for this reason, the dissertation proposes three different solutions to plan the deployment of PV system in three different contexts. Each solution has its own implementation which takes into account the constraints and the opportunity of every single context. However, they share a common goal: to provide an optimization of the placement of the PV panels and to provide an economic analysis that can help those actors to evaluate the cost and the benefits of such a system.

The thread that runs along the dissertation is that all the proposed solutions have been developed to be used in a co-simulation environment. Co-simulation is a technique that allows studying the interaction among the simulation of different aspects of a complex system. This technique is particularly useful in the Smart Grid field, where there are a lot of different actors that are involved and interact with each other.

Therefore the solution proposed in this dissertation have various functionality for researchers: i) the proposed solutions can be used in complex cosimulation scenarios to provide a communication infrastructure for the Smart Grid ii) the proposed smart meter implementation can be used to test new technologies or policies, iii) the PV frameworks can be used to assess the impact of the new actors and their distributed energy plant on the electric network. However, also operators and customers can use the proposed solution as a standalone product to evaluate the cost and the benefits of monitoring the grid infrastructure or the deployment of PV systems.

Contents

List of Figures

List of Tables

Acronyms

3SMA 3-phase Smart Meter Architecture

AC Alternate Current

AMI Advanced Metering Infrastructure

API Application Programming Interface

BAU Business As Usual

BEMS Building Energy Management System

COE Co-simulation Orchestrator Engine

DRTS Digital Real-Time Simulator

DSO Distribution System Operator

DAQ Data AcQuisition board

DB Database

DC Direct Current

DeC Device Catalog

DER Distributed Energy Resource

DSPL Domain-Specific Programming Language

DSM Digital Surface Model

EA Energy Aggregator

EHP Electric Heat Pump

ESI Energy Systems Integration

FMI Functional Mock-up Interface

fdir Fault Detection Isolation and Restoration

FLISR Fault Location Isolation and Restoration

GHI Global Horizontal Irradiance

GIS Geographic Information System

GPPL General-Purpose Programming Language

HIL Hardware-In-the-Loop

HTTP HyperText Transfer Protocol

ICT Information and Communication Technologies

IoT Internet of Things

IPR Intellectual Property Right

LV Low Voltage

MES Multi-Energy System

MAN Metropolitan Area Network

MQTT Message Queuing Telemetry Transport

MV Medium Voltage

PHIL Power Hardware-In-the-Loop

PTP Precision Time Protocol

PID Proportional Integral Derivative

PMU Phasor Measurements Unit

POD Point of Delivery

PT Payback Time

PV PhotoVoltaic

RMS Root Mean Square

RES Renewable Energy Sources

REC Renewable Energy Community

REST REpresentational State Transfer

ROI Return On Investment

SE State Estimation

SOC State Of Charge

TCP Transmission Control Protocol

UDP User Datagram Protocol

VCS Voltage Control System

VRS Voltage Regulation Service

Chapter 1

Introduction

1.1 Smart Grid

The legacy electric grids that have been used for the past 70-80 years have a "linear architecture": large plants generate all the electrical power and feed it to the high-voltage part of the system known as the "transmission network". The transmission network is used to transport the energy for long distances to the various substation closer to the end user. From these substations than the distribution network takes care of providing energy to the customer. The different actors of the transmission system communicate only basic information to each other to ensure the correct functioning and the automation is extremely simple and usually designed for well-known malfunction cases. The distribution has instead even simpler functionality and the communication is mainly used for billing purposes. However, as shown in Figure 1.1 the aim of the future Smart Grid is to move away from this structure towards an approach where the actor of the grid can exchange both information and energy with each other. The evolution in the field of Information and Communication Technologies (ICT) is fostering this transformation. In particular, ICT offers the possibility to increase the monitoring capabilities in the grid ensuring more reliable operation and cost-effective use of the resources. By using these technologies the aim is to transform the electric grid into a Smart Grid, capable of exchanging various kinds of information among the different actors at each level of the grid. Thanks to such a system the grid will be capable of promptly reacting to various situations

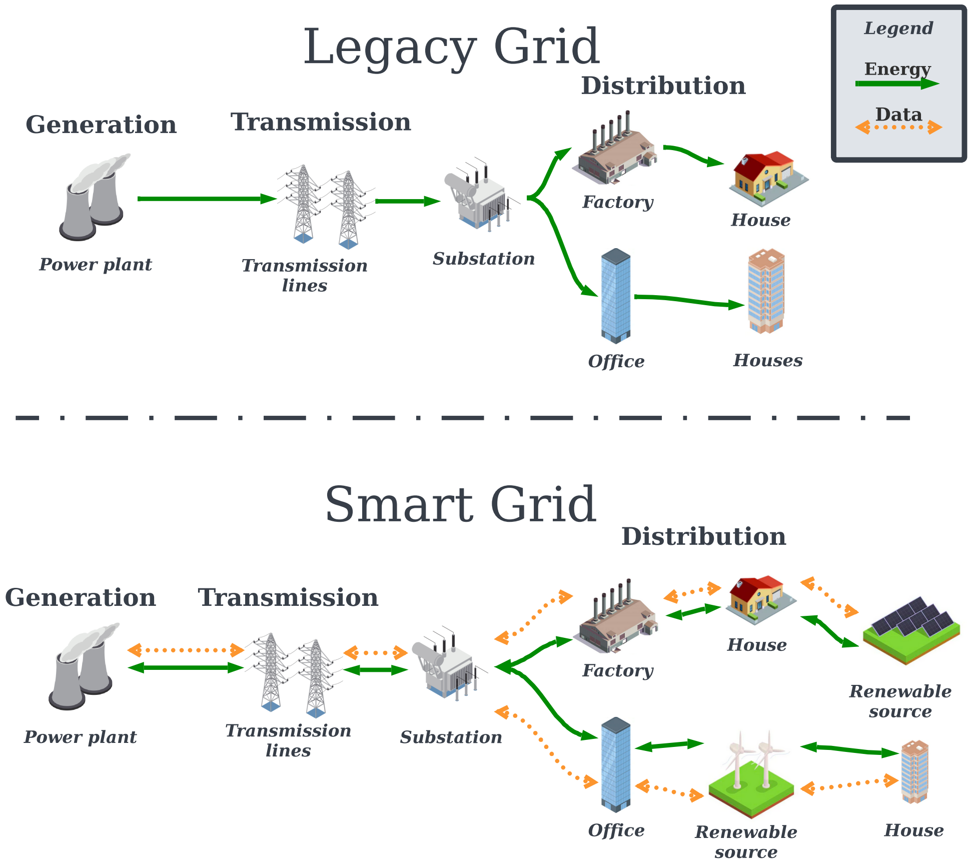


Fig. 1.1 Comparison of the legacy grid vs Smart Grid

that may occur in the network and, whenever possible, even take an autonomous decision.

Another factor that is pushing the transformation of this traditional paradigm is the urgent necessity to move from fossil-fuel generation toward a more sustainable energy source. Renewable energy plants, therefore, will be more and more diffused however unlike their legacy counterpart they will be smaller but more distributed. In this scenario, it will be essential to have deeper and more precise monitoring of the grid to ensure its correct functionality. In addition to that due to the uncertainty of the output of some renewable energy sources, such as solar or wind generators, it will be essential to constantly monitor the grid to ensure that the supply of the Smart Grid meets the demand of the users. In the Smart Grid scenario also the user will have a different role and many of them will become prosumers. The prosumer is a user which owns a small private power generation plant to produce a portion or even the entirety of the energy that they consumes. Prosumers generally own some kind of renewable energy source, such as photovoltaic systems, that is capable of generating a relevant amount of energy that creates 3 possible situations: - the energy produced meets exactly the prosumer demand - the system produces more energy than needed. - the system cannot satisfy the user's demand. However, these situations are not mutually exclusive: during the day there may be moments when any of them occurs. Therefore in the Smart Grid we will have a bidirectional flow of energy that will be withdrawn or injected into the grid by the prosumers, according to the need of the moment. Overall Table 1.1 summarizes the main differences between the legacy grid and the Smart Grid paradigm, which are [8]:

- Power generation: Centralized vs. Distributed

  In the legacy grid the energy is generated in few, big and centralized plants thus reducing the possibility to include smaller but numerous and dispersed plants. Smart Grids instead will be capable of integrating such sources, especially renewable energy sources, which can help to balance the load during the peak our

- Communication: One-way vs Two-way

  Thanks to the ICT technologies the different actors (hardware and software) will be capable of communicating at any level of the grid (generation, transmission and distribution)

- Metering: Basic vs. Smart

  Smart sensor will be deployed in the grid. These devices will be not only able to communicate measurements to the other actors but these could be also capable of executing non-trivial algorithms to support the monitoring and the resilience of the grid

- Power flow: Unidirectional vs. Bidirectional

  As mentioned before the presence of distributed energy sources owned by prosumers will require the possibility to both withdraw and inject power into the grid not only to satisfy the energy demand of the customer but also to ensure a proper load balance across the grid.

- Restoration: Manual vs Automatic

Nowadays human intervention is needed in case of fault to detect its location and repair it. In the Smart Grid, also thanks to the smart sensor and the detailed monitoring, the grid will be capable of self-healing and temporarily restoring the service. In addition to that the sensor will be also capable of indicating the precise location of a fault to speed up the restoration of the grid to the normal operational condition

- Actuation: Limited vs Complete

  Nowadays the companies that manage the electric network have control of the network substations but the control and access to the network itself are quite limited. In the Smart Grid, together with the sensor, also smart actuators will be deployed to ensure a more granular control on the grid from the generation plant to the final user

Characteristics	**Legacy grid**	**Smart grid**
Power generation	Centralized	Distributed generation
Communication	One-way	Two-way
Metering	Electromechanical	Smart digital
Monitoring	Manual	Self-monitoring
Power flow	Unidirectional	Bidirectional
Architecture	Radial	Network
Restoration	Manual	Self-healing
Controlling	Limited	Pervasive

Table 1.1 Comparison among the characteristics of the legacy grid and the Smart Grid

However this shift towards a Smart Grid requires huge investment due to the large size of the electric grid, thus it requires careful study, design and planning to justify its cost. Given the number of technologies and expertise that will be involved in the Smart Grid, the research has been focused on various and different topics. In [9] nine main research areas were identified:

1. Smart Grid reliability and awareness:

   In the Smart Grid context, the aim is to quickly respond to, or even forecast, possible situations that may cause harm to the grid itself or just compromise the efficiency of the service. The aim is to have a self-healing grid by using smart devices and control procedures that can at any time provide detailed monitoring and automatically ensure the correct operational status of the grid.

2. Energy efficiency

   The aim is to use strategies and provide incentives to push the customer to use the energy efficiently, for example by reducing consumption during the peak hour. Strategies like Demand Response, for example, do this by trying to continuously match power supply and demand.

3. RES

   As the need to find alternatives to fossil fuel generation increases, more and more renewable energy sources will be deployed in the grid. However due the efficiency of most of these technologies strongly depends on external factors thus making them not completely predictable. Therefore there is a need to find the correct placement of such power plants to ensure their efficiency and maximize their usefulness for the grid.

4. Energy storage

   The storage of power is a big challenge, the aim is to store the electric energy as such in large-sized batteries or in other forms of energy (such as thermal energy). This aspect is heavily influenced by the usage of renewable energy sources which can produce more energy than the actual demand of the grid. Another important factor is the rapid diffusion of electric vehicles which both increase the demand and could be even used as temporary energy storage itself.

5. Transmission

   The deployment of RES in distributed plants will reduce the amount of energy that needs to be transferred across the grid and will also reduce the distance that this needs to travel. Also in this case the presence of electric vehicles will enable new strategies such as the usage of such cars as movable batteries to transport energy.

6. AMI

   The usage of smart electric meters aims to increase the monitoring capabilities of the grid and will provide more direct communication and control from the power plant to the final user. In this field, the Internet of Things paradigm plays a crucial role since it introduces various protocols and techniques to enable the deployment of such metering infrastructure.

7. Distribution

   While nowadays the distribution level is monitored mainly for billing purposes, in the future smart appliances and meters will be used to ensure more efficient usage of energy and increase the reliability of the last mile of the grid.

8. Security

   Since most of the features of the Smart Grid are strictly related to ICT communication and infrastructure, the security and privacy of the communication will be of crucial importance. For this reason, it may be necessary to design and develop new domain-specific solutions to ensure it.

9. Network communication

   Given the huge amount of information that will be transferred across the Smart Grid, both the infrastructure and the protocols available right now could be not suited for that scenario. Therefore there is a need to carefully analyze the challenges and find effective solutions to support the communication in the grid at all its levels (generation, transmission, distribution)

My Ph.D. activity focused on two of that topics AMI and RES, aiming to solve the following question:

- About the AMI
  - Which are the objectives of an AMI?
  - Which IT solutions are the most appropriate to reach such objectives
  - Which feature should an AMI have?
- About the RES
  - Why PV seems to be the most diffused RES in the upcoming future?
  - How to optimize the production of a PV system?
  - Which are the upcoming scenario in which PV panels will be used and how can those be addressed?

Due to the complexity of the Smart Grid scenario, other questions raise up:

- which research tools can be used ?
- What is the cosimulation and how can be useful for the Smart Grid?
- How to integrate AMI and RES with cosimulation?

The following section will give an introduction to these three topics analyzing their characteristics, the opportunity and the challenges that they present.

1.2 Advanced Metering Infrastructure

Nowadays the meters in the customer's houses are mainly used for billing purposes and, despite some exceptions, the reading of the energy consumption still needs to be done manually. With such limited capabilities therefore it is impossible to ensure proper detailed monitoring for the operators of the electricity distribution system and for the customer itself. Even at the transmission, the capabilities of the meter are extremely and in case of fault and outages the automation mechanisms are extremely simple and the restoration of the service needs human intervention. However, as already mentioned previously, a proper Smart Grid will need a constant and widespread monitoring mechanism to increase its reliability and enable new control strategies and services. ICT, and the Internet of Things (IoT), in particular, will be extremely important to achieve such objectives. The IoT approach provides various tools and technologies which can be used to connect to the internet the different sensors and actuators that will be deployed on the grid, enabling them to exchange data and to be remotely controlled. As shown in Figure 1.2, with this approach the sensing devices of the grid, commonly known as meters will become *smart meters*, such devices will be able to collect and send data, process them, and even take autonomous decisions for the management of the smart grid. Smart meters are usually composed of a data-acquisition board and a computation and communication board. The *data acquisition board* is in charge of collecting the analog measurements of voltage and current and transforming them into digital signals for further processing. The sampling frequency of this board may vary according to the purpose of the meters. According to the literature, the data acquisition board of a

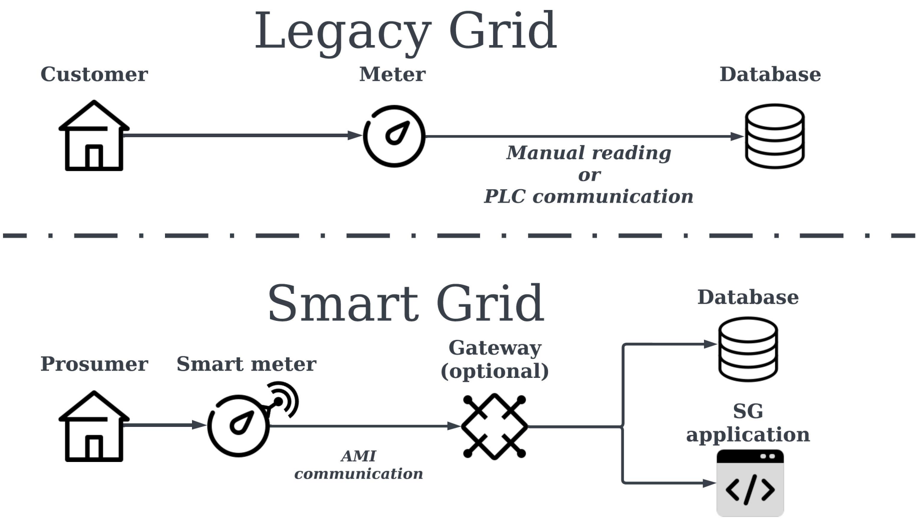


Fig. 1.2 Differences among the legacy metering infrastructure and the Advanced Metering Infrastructure of a Smart Grid

proper smart meter should respect a set of minimum requirements, it particular a meter must be able to:

- collect measurements with a level of accuracy that may depend on its application (**Accuracy**).
- send and/or receive commands to/from other devices (**Control**).
- be able to receive updates that improve its functionalities(**Updates**)
- obtain a precise time reference to perform their operations since time synchronization is a crucial aspect in the field of electric energy (**Syncronizathion**)
- show information regarding the monitored data (optional in case of meter not devoted to the customer service(**Display**)

The second component is the one that makes the meter "smart" since it enables bidirectional communication between the meter and the rest of the grid and, when needed, can process the meter's measurement. This data processing can be useful to run onboard algorithms that can be simple statistics or more complex grid management algorithms. This component allows the meters to be part of a communication infrastructure, which is needed to receive and send any kind of data and commands. For such communication many standards and protocols are already available, however, multiple aspects need to be considered to select correctly:

- Data volume: various smart devices (not only smart meters) will be deployed on the Smart Grid therefore a high amount of data will be generated and those data need to be transferred
- Confidentiality: due to the sensitivity of some of the data that will be monitored it is essential to provide a secure and safe communication among the AMI actors
- Cost effectiveness: since the objective is to deploy a lot of devices to ensure a deep network monitoring, the cost of the devices and the communications infrastructure need to be taken into account.
- Upgradability: the devices should be able to receive both software and hardware upgrades without major issues.

In addition to that during, the design phase also the strategies and the technologies to use need to be carefully selected. For example, in some applications, gateways or concentrators could be used to filter and reduce the amount and the size of the data to transfer. Various communication topologies can be used, multiple technologies can be selected, such as:

- Power Line Carrier (PLC)
- Optical fiber or copper
- Cellular line (GPRS, 3G, LTE, 5G)
- Bluetooth
- Zigbee
- Internet

- Satellite

It is important to underline that two main types of smart meters can be used in the Smart Grid: in addition to the one that is the direct evolution of the meters that we can find in each house, also the so-called Phasor Measurements Unit (PMU) will be deployed. PMUs are extremely precise meter that allows monitoring the grid at the transmission level, hence these devices are particularly useful for the operator of the transmission system since they can provide a detailed overview of the operational status of the grid.

Therefore the usage of smart meters and PMUs within an AMI enables a series of other features and services at every level of the grid, which nowadays is nearly impossible such as [10]:

- Real-time pricing for the final customer
- Power quality monitoring
- Fault and outage autonomous detection
- Demand response policies
- Detection of energy theft
- Prosumer energy management

These kinds of functionalities will bring sizable benefits both to the final customer and also to the energy suppliers and the operators of the transmission and distribution system, those are summarized in Table 1.2 [11]

As summarized in the table, we can notice how AMI can also facilitate the deployment of RES. Since the production of most of the RES heavily depends on atmospheric conditions (sun presence, wind strength, rain, to name few) the presence of an AMI is necessary to ensure a correct power balance across all the Smart Grid. Since the RES will be more and more diffused, for reasons explained in Section 1.3, RES and AMI pose challenges that need to be faced by developing proper metering infrastructure.

The Ph.D. activities focused on this aspect, my research aimed to provide a reliable communication infrastructure and a smart-meter prototype to compose a

Suppliers and grid operators advantages	**Customers advantages**	**Common advantages**
More precise, more frequent and cheaper measurements	Increased consciousness of electricity usage	Reduced energy wasting
Possibility to manage peak demand through detailed Demand-Response policies	Possibility to reduce bills by following Demand-Response plan	Increase service reliability during peak hours
Easier planning of generation and maintenance	Possibility to customize electricity usage	ensuring network resilience and improve service quality
Easier deployment and management of RES	Possibility to self-generate energy through small RES installation	Increase energy supply and reduce fossil-fuel generation
Possibility to offer and sell new services	Increased electric vehicle support	Increased energy efficiency and reduce CO_2 emission

Table 1.2 Overview of the advantages provided by AMIs

proper AMI architecture. This infrastructure, together with the smart meter, should provide a foundation on which to build the functionality of the Smart Grid, such as automatically reacting in case of fault or providing an updated estimation of the operational status by exploiting the monitoring and processing capabilities of the smart meters.

1.3 Renewable Energy Sources

With the agreement signed in Paris in 2021, more than 190 parties committed to keeping the rise in the global average temperature below 1.5 Celsius compared to pre-industrial measurements. In addition to that the past 3 years, due to the outbreak of COVID-19 and the military escalation in Eastern Europe, underlined the still strong dependency of society on fossil fuels. The energy cost skyrocketed, putting in serious difficulty the entire industrial sector and private customers. Indeed, as shown in Fig 1.3 more than 80% of the energy production uses fossil fuels such as oil coal, or gas. However, these two challenges can and should be faced together: it

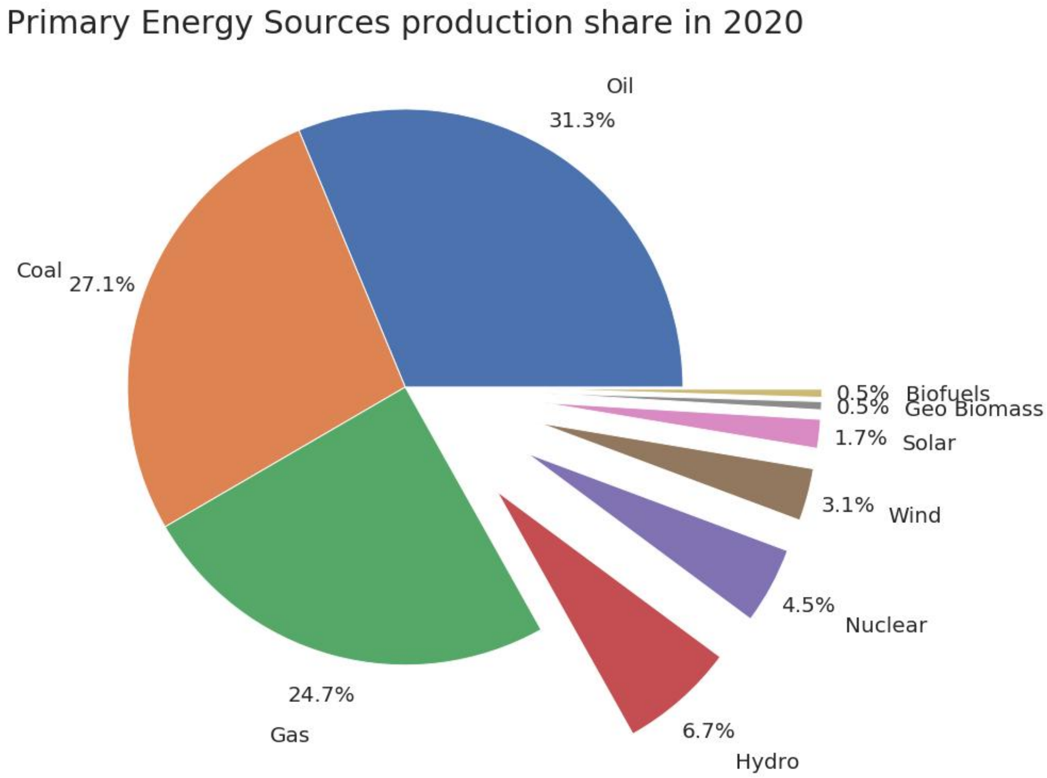


Fig. 1.3 Percentage of the source used for energy production in 2020

is not feasible to keep investing in inefficient energy production and distribution that can even harm the future of human beings. RES can instead improve the efficiency and the resilience of the grid while helping in reducing the emissions of CO_2. This transition requires a serious change in the ways in wich the energy is produced and consumed, we need to increase:

- the percentage of energy produced through RES
- the energy efficiency
- the ectrification of the products sectors (for example electric vehicle and heat pump)
- the usate bioenergy and hydrogen
- the diffusion of energy storage systems

Among these different possibilities, the production of energy through RES is nowadays the easiest and cheapest solution. Therefore, in most of the countries, RES are concentrating most of the investments devoted to rise the capacity of electric generation.

As shown in Fig 1.4, among the different technologies, the usage of photovoltaic panels is one of the most common choice in the past year since its cost lowered by around 85% in the last ten years. This technology alone is concentrating around 50% of the investment dedicated to RES around the world[12]. In the same figure, we can see how wind generation is a common choice of investment. This happens because they allow having large-sized power plants where a lot of energy can be generated and injected into the grid. This solution is particularly easy to adopt even for a traditional grid where the standard topology assumes to have few large power plants that produced all the energy needed to satisfy the customer's demand. However, as mentioned before in the Smart Grid this will be not the only solution, small and distributed power plants will become more common [13–17].

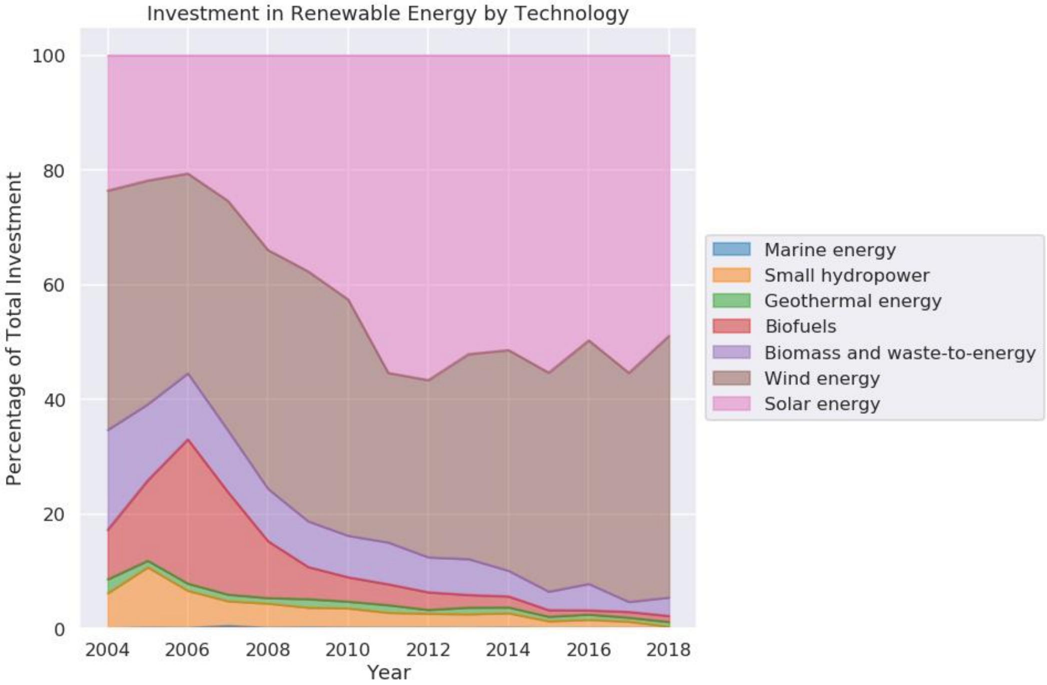


Fig. 1.4 Evolution of the investment dedicated to the research and development of renewable energy technologies

1.3.1 REC and PV installation

A Renewable Energy Community is an association that uses RES technologies to balance the energy supply and demand of its residents, to oversimplify we can see a REC as a group of prosumers with a shared energy production system. RECs play a crucial role in successful energy transitions as they increase the flexibility and resilience of energy systems, and spread the acceptance of bigger renewable energy projects [18]. RECs are composed of a variety of actors (such as households, public institutions, business activities, etc.) sharing the common goal of producing energy from renewable sources to maximize self-consumption, hence reducing the need to withdraw energy from the electric grid. However, a simple installation of a shared renewable energy plant through an investment of a community is not enough to set up a REC. Many other challenges need to be taken into account, such as legal aspects, grid connections, demand flexibility policies, and storage systems [19]. The legal institution of a REC is the first challenge that needs to be analyzed since regulations vary from place to place. For example, EU and North America have distinct policies [20, 21] with differences even among countries [22–24]. In addition to that, due to the complexity of the system and the variety of actors involved, RECs need careful planning to ensure efficiency and success [25]. The first thing to consider is the type of renewable energy that should be deployed. As previously mentioned, cost reduction makes PV systems the perfect choice for a collaborative approach. For such reason, the PV solution will become even more widespread since it is, among the RES technologies, the one that does not require a huge amount of space for deployment and PV panels can be installed even in urban environments without any major difficulty.

Due to those characteristics, PV panels are particularly suited for the needs of the prosumers, those customers that deploy their own small-sized PV system to partially satisfy their own energy demand. Similarly, it is also an effective solution for the REC. However, while one could think that the easiest solution could be to deploy as many PV panels as possible this approach is not practical. Despite being cheap the cost deployment and the maintenance of PV panel is not negligible so for each scenario it is needed to have careful planning. The design of a RES system, whether it uses PV or other technologies, needs to take into account the trade-off between cost and benefits and try to optimize the investment. Therefore there is a need for tools that can simplify ad optimize the planning phase, which allows simulating as realistic as possible the different scenarios to provide effective renewable energy systems.

This indeed was the objective of my activities during the Ph.D, my research focused on the development of a framework for the optimization of the PV placement in the context of prosumers, EA and REC. The framework is able to provide optimal PV configuration of the whole system, provide economic analyses for the stakeholder of such systems and evaluating the environmental benefit related to the usage of renewable energy.

1.4 Cosimulation for Smart grid

Simulations are widely diffused in all fields of research since this help to test the reference case study of different scenarios that may be hard or unfeasible to reproduce in the real world. This is the case also for electric network research where models are used to reproduce and study the behavior of the various components of the grid in a variety of situations. However, this approach is extremely difficult when applied to the Smart Grid. Due to the number of actors involved and the variety of technologies used, it easy extremely hard if not impossible to create a unique model that synthesizes the complex interaction and evolution of the grid. The two main limitations of this approach are:

- Expertise

  Given the variety of topics, a single model developer cannot possess the knowledge needed to design a complete model of the whole Smart Grid in all its facets

- Tools

  Nowadays there is plenty of software to have detailed models of the different parts that compose the Smart Grid, however, each of them is extremely specific and mainly suited for a single technology. There is still no universal tool that can model all the different components of the Smart Grid and their interactions.

- Hardware

  While the computational capabilities of modern hardware are impressive and continuously improve it is still unfeasible or even impossible to run complex

> simulations that would be needed to study the Smart Grid behaviors. In addition to that some actors and devices of the Smart Grid may require custom hardware which is often not capable of running other kinds of simulation.

To solve those limitations in the study of Smart Grid the technique of cosimulation has been adopted. Co-simulation, also known as co-modeling or co-simulation framework, refers to a methodology or technique used in computer simulation to integrate and coordinate multiple simulation models or software tools, allowing them to interact and exchange information during the simulation process[26].

In a co-simulation setup, different simulation models, which can be developed using different programming languages or simulation environments, are combined to create a more comprehensive and realistic simulation of a complex system. Each individual model represents a specific aspect or component of the system, and the co-simulation framework enables these models to communicate with each other and simulate their interactions.

The co-simulation approach is often used when a single simulation model cannot adequately capture the behavior of a complex system due to its size, heterogeneity, or interdependencies between different components. By integrating multiple models, each specialized in modeling a particular aspect, co-simulation enables a more accurate representation of the system's behavior and interactions.

Typically, co-simulation involves the exchange of inputs and outputs between the individual models at predefined synchronization points during the simulation. The models run in parallel, and at each synchronization point, they exchange information, such as state variables, control signals, or data packets. This exchange enables the models to react to each other's outputs and update their own states accordingly, simulating the dynamic behavior of the overall system.

Co-simulation finds applications in various fields, including engineering, physics, computer science, and social sciences. It is commonly used in areas such as automotive engineering, aerospace, power systems, communication networks, and even urban planning. The key advantage of co-simulation is its ability to integrate specialized models and leverage their individual strengths to provide a more comprehensive understanding of complex systems.

Cosimulation is an effective tool to study systems with such complexity and variety as the Smart Grid, however, it does not come without challenges to solve. The first major challenge is time synchronization since each simulation tool may have different timing requirements related to how rapidly the simulated system evolves. For example, while the simulation of the energy production of a PV system works fine with 15 minutes time-step, the simulation of the grid behavior in case of a fault requires a time step of only $25 \mu s$. Another aspect of this same challenge is the different ways in which simulators can work: event-based vs. time-based. In event-based simulations, such as the simulation of control algorithms or network communication, the time interval between 2 subsequent events can vary by an unknown amount of time. On the other hand, time-based simulation works with a fixed time-step, which means that all the events of the simulation are equally spaced in time.

The cosimulation approach reduces the need of having a single extremely powerful device to study a smart grid scenario, instead using this approach a researcher is able to connect together multiple devices, each responsible for running a single simulation. While still, some specific simulations require ad-hoc hardware to be performed in a lot of cases simulations can run on desktop PC or even laptops.

Moreover, it allows using Hardware-In-the-Loop (HIL) approach to test such hardware in multiple scenarios that could be dangerous or unfeasible to reproduce in the real world. Therefore cosimulation techniques have been widely used by researchers in all the fields of the Smart Grid as summarized in Table 1.3.

During my PhD cosimulation has been studied and used in the work described in 2 and 8, both these works used the HIL technique and in 8 also deals with simulators with different time requirements.

1.5 Proposed scientific novelties

The first contribution of this work is related to the field of the AMI. A preliminary study of the literature underlined a lack of flexibility in the proposed infrastructure and meters. The AMI solutions proposed in the literature are usually strongly related to a particular challenge that needs to be solved such the usage of a particular technology or the strengthening of the security of the communication network. While those approaches are valid that there also the need to provide a more flexible infrastructure that enables the integration of various solutions, both hardware and software by

Field	**Advantages**
Smart Grid reliability and awareness	Cosimulations has been used to test the resilience and assess the performance of the grid, by testing monitoring and control platform and to analyze the interaction between the smart grid and the ICT infrastructure.
Energy efficiency	Cosimulation has been used to study user behaviors in different conditions, to test Demand-Response policies, and more in general to evaluate possible solutions to improve the efficiency of the Smart Grid.
RES	In this field, it has been used to analyze the deployment of renewable energy plants and their interaction with the grid and its components such as energy storage solutions.
Energy storage	It is extremely useful to analyze and test the hardware or the different policies that could be applied to store the energy. Cosimulation is widely used to analyze the interaction between RES plant and the energy storage solution while also considering the rapid diffusion of electric vehicles.
Transmission	Cosimulations have been used to test control algorithms and devices for the transmission grid or to test FLISR solution and other self-healing strategies.
AMI	Cosimulation has been extremely useful to analyze the impact of a large number of smart-meters on the Smart Grid while taking into account all the ICT aspects involved in the communication of a large number of connected devices.
Distribution	Cosimulation has been exploited to find solutions to improve the service for the customers and to test effective solutions to manage the interaction between the prosumers and the Smart Grid.
Security	Cosimulation allows analyzing the interaction between the electric and the communication network, to test their resilience to cyberattacks and more in general to identify the weak points that malicious third parties could exploit.
Network communication	is one of the most common topics where cosimulation is needed to test the integration of ICT in the electric grid. It is extremely useful to test communication technologies and protocols or to identify possible bottlenecks in the communication infrastructure which can undermine the efficiency and the security of the Smart Grid.

Table 1.3 Usage of cosimulation for the Smart Grid research

defining minimum requirements that can be easily satisfied. In addition to that, while there are various solutions for the monitoring of the grid at the transmission level, the meters used for this purpose are lacking most of the features that the Smart Grid requires. Therefore the first major achievement of the Ph.D. activities was the design and the development of AMI and a smart meter prototype capable of filling these gaps. The proposed metering architecture uses common ICT technologies enabling easy integration of third parties tools, it also includes a mechanism that allows the self-configuration and the auto-update of the devices deployed on the smart grid. Together with that, we proposed a smart-meter prototype that uses cheap and open-source hardware and software. This smart meter is capable of collecting 3-phase measurements and features the possibility to be used for Edge computing and Fog computing.

The second major contribution of my research activities is related to the field on REC, in particular in the usage of PV systems in the REC context. This is a very active field of research as the energy transition is becoming a crucial objective to achieve in the immediate future. The literature identified the REC paradigm as one of the most effective ways to increase the diffusion of renewable energy sources and various solutions are proposed. Most of the works study the interaction among different energy sources or are dedicated to the design of energy storage solutions. However, we noticed that there is a lack of work that tries to optimize the energy production of renewable sources. Therefore the second novelty proposed by my activities aim to provide an optimization of the the PV systems in a REC scenario. This result was achieved proceeding by steps. The first work described in Chapter 3 aims to design an algorithm for the optimal placement of PV modules in order to maximize the production of the system. In order to do so we designed a framework that uses Geographic Information System (GIS) technologies to provide a realistic estimation of the production of a PV system. The second work proposed is built upon that but instead takes into account the main opportunity offered by the REC paradigm: the sharing of the assets. Therefore this solution considers the roofs of the participants of the community as a common resource therefore the energy produced by the system could be shared among them. The third and final work makes an even further step. Up until now we were able to provide a realistic and detailed estimation of the energy produced by a shared PV system, however during the planning phase also the energy demand of the community should be considered to better evaluate the possible benefit that could be obtained thanks to the REC approach. Therefore this third work provides a complete framework that:

- provides an optimal placement for the PV modules on the roofs of the community
- provides a realistic estimation of the energy demand of the community
- uses the previous results to evaluate the possible economic and environmental benefits that the REC would obtain

1.6 Structure of the dissertation

The structure of the dissertation is the following. Chapter 2 provides the description of the design and development of 3SMA, a 3-phase Smart Metering Architecture. The chapter will describe the challenges to face and compares the proposed platform w.r.t. other literature solution. The result of this work is a flexible communication platform for the Smart Grid and a configurable smart meter prototype that can be used together with it. Chapter 3 gives an introduction to photovoltaic technology to underline its functionalities and its limitations. Then it provides a brief description of the economic aspect of the renewable energy sources and their cost, with particular attention to the solutions that exploit the PV technologies. Chapter 4 proposes a framework that includes a placement algorithm and an economic analysis that aim to optimize the installation of PV panels for a prosumer. This is the first work that introduces the foundational approach and tools that will be used in the chapters after that. Chapter 5 proposes instead a similar framework but with the approach of the Energy Aggregator EA. This new approach includes new constraints and opportunities that are required to modify the previous solution. Chapter 6 finally proposes a framework for the planning of a REC, by providing an estimation of the balance between its energy supply and demand, while evaluating both the economic and the environmental benefit that it would obtain. Chapter 8 presents a co-simulation infrastructure that includes the output of the work described in Chapter 2 and Chapter 3. This chapter illustrated a realistic case study to evaluate the effect of a prosumer PV system on a Smart grid, this allows for easily testing of different policies and tools which can bring benefit to both the actors.

Chapter 2

3SMA

2.1 Introduction

Power networks have evolved in the last decade to reach a new level of efficiency and reliability, and to include new and pervasive RES. To achieve these targets, smart grids are now implemented at all levels, from the power plants down to end-user networks. A modern grid does not only include wires, substations, transformers, and other electrical devices, but it also exploits the Internet to better manage the energy. In other words, a modern smart grid is an electrical grid that adopts strategies coming from the ICT area to autonomously gather information and improve its efficiency, reliability, economics, and sustainability.

Among the different devices that are possible to connect to a power grid, next-generation meters (also known as *smart meters*) implement IoT functionalities to collect data from the power grid and to communicate with other hardware and remote software entities over the Internet. The resulting AMI [27] offers different features to each actor playing in the energy marketplace [28].

For example, Distribution System Operator (DSO)s can use smart meters to increase the efficiency of the distribution network and to improve its capability and reliability. Retailers can use smart meters to forecast the variations of the network load to create adequate policies to reply to demand. Prosumers can use smart meters to control their energy production (highly variable due to the intermittent behavior of renewable sources) and utilization and they can reduce the cost of their electricity bills [10].

Unfortunately, modern metering infrastructures have to face multiple challenges to be cost-effective, scalable, and reach the desired penetration on the market. Among these hurdles, we would like to recall the cost of the communication infrastructure [29], the price of the hardware devices [30], and the extremely high intricacy of modern networks. To propose cost-effective and scalable solutions, we thus need to deploy low-cost devices and to be able to configure, control, and update all units of the network remotely. For example, a self-healing grid should be able to automatically reconfigure itself in case of an outage, automatically performing fault location and service restoration [31]. Unfortunately, this process is still partially executed manually nowadays. This approach reduces the reliability of the network, increases the outage time, augments the network dependability from human intervention, and often results in customer dissatisfaction and extra costs for the system operators. To design a true self-healing grid, all previous process steps should be automated, minimizing human intervention. Distributed RES also need particular attention. RES breaks the traditional schema of the unidirectional energy flow as they can be installed by end-users (the so-called prosumers), which can both consume and produce energy, and are also quite unpredictable. For these reasons, State Estimation (SE) algorithms are required to help the operators to monitor the state of the grid. SE strategies evaluate the electrical parameters of each node and build a comprehensive picture of the grid to prevent possible contingencies,

To overcome some of the above challenges, we propose two main contributions:

- We describe a distributed software infrastructure, including an advanced metering technology, providing several benefits. This infrastructure admits bidirectional communication for the smart units and it enables the devices deployed on the grid to self-configure and auto-update themselves. Moreover, it allows the information related to the deployed devices to be used by the remote services included in the system. In this way, we enable interoperability with third-party software, which could facilitate further general-purpose services and business cases.

- A design pattern for an IoT-enabled 3-phase smart meter. Our design defines a 3-phase meter with the capability to run multiple algorithms, either on-board, on-fog, or on-cloud. Thanks to the self-configuration and auto-update procedures, any algorithms can be added, updated, or removed on the fly without affecting the rest of the system. This makes the whole solution flexible and opens it to future improvements of the smart grid technology. In our view, this feature is essential to unlock new services and to foster new business opportunities for the actors playing in the energy marketplace.

During our design process, we pay particular attention to the software engineering phase and to the meter (hereinafter, referred to as 3SMA) design stage. We present an evaluation of our distributed infrastructure, and more specifically of our 3SMA, based on real-time simulations. Our simulations proved that our architecture imposes very low requirements for the device needed to build the 3SMA, thus reducing the cost and making the entire structure highly scalable.

2.1.1 Contributions

To summarize, our work includes the following novelties. We design and develop a scalable distributed software infrastructure for advanced metering presenting the following features:

- It provides bidirectional communication between smart devices and novel services.
- It enables cloud, fog, and edge computing to support new smart services to manage the grid.
- It unlocks new decentralized services and improves the quality and the performances of those already existing.
- It allows the self-configuration and the auto-update of the smart devices installed across the smart grid.
- It provides remote services with (near-) real-time information about both the smart devices and the smart grid status.
- It integrates third-party software and enables their interoperability with all the entities, either hardware or software, in our solution.

We also propose a novel design pattern to develop our 3SMA, i.e., a new 3-phase meter, which:

- Embeds IoT functionalities.
- Is compliant with the requirements of the proposed distributed software infrastructure.
- Provides bidirectional communication with the other actors in the infrastructure, either hardware or software.
- Allows remote grid data samplings and transmissions in (near-) real-time.
- Runs custom algorithms for advanced smart grid management, either on-board or distributed, thus, exploiting the offered cloud, fog and edge computing capabilities.

2.1.2 Roadmap

This chapter is organized as follows. Section 2.2 focuses on related works and their main differences from the approach proposed in this chapter. Section 2.3 first describes the proposed distributed software infrastructure for advanced metering, introducing both the identified actors, and the technologies adopted to enable the communication and the data exchange among them. Then, it presents the proposed IoT-based 3-phase smart meter (i.e., our 3SMA) with its main characteristics and interactions with the other entities in the distributed software infrastructure. Finally, Section 2.3 reports the on-cloud, on-fog, and on-edge applications used to test and assess both the 3SMA and the infrastructure. Section 2.4 concentrates on the communication flows among the identified actors, either hardware or software, in our solution. Section 2.5 describes the case study and the experimental setup we used to evaluate our infrastructure. Section 2.6 includes our experimental analysis of the metering infrastructure. To conclude, Section 2.7 reports our final remarks and some hints on possible future works.

2.2 Literature review

As introduced in Section 1.2, advanced metering infrastructures for smart grids enable bidirectional communications to monitor the energy transmission and distribution process [32]. The last few years have seen growing attention on the area, focusing on different approaches and distinct technologies to reach advanced management architectures. Different standards and protocols are diffused and have been proposed such as DLMS/COSEM [33], SML [34] and IEC 61850 [7][35][36], these solutions aim to address different challenges of the Smart Grid scenario and are designed for different actors according to their needs.

Table 2.1 shows some of the main IEC standards involved in the Smart Grid communications which are more than 100. This variety is related to the diversity of actors involved in the Smart Grid. As shown in Table 2.2 the bandwidth and the latency requirements can be quite different according to the application target and its criticality. Finally, Table 2.3 shows the main IoT protocol that has been used in literature for the Smart Grid communication. It is important to notice how the characteristics of each protocol make it preferable over another for a particular application. While this approach ensures a higher level of optimization, this also requires careful testing and verifying, case by case, which is the best choice according to the requirements of the application. Therefore interoperability is not a limit but it is instead a feature that needs to be taken into account during the design of an application.

Thus interoperability plays a crucial role in the design of AMI and smart meters that need to offer multiple ways to communicate with the grid.

Meloni et al. [37] introduce a new infrastructure to guarantee device interoperability. They also introduce the possibility to develop additional cloud services by providing proper API to obtain information from the meters. However, the suggested infrastructure manages only the distribution network and not the transmission one. Moreover, the authors design their meters as sensing devices able to collect and send data, but unable to perform any sort of data processing.

Chen et al. [38] illustrate a smart-metering architecture for IoT-based meters. They design it using edge computing to overcome the problems related to the latency and the bandwidth, typical of cloud computing. However, their architecture does not provide cloud or fog computing, which, in our opinion, are crucial to unlocking advanced and third-party services, for smart grid management.

Yan et al. [39] implement their infrastructure adopting fog computing. They perform data processing on clusters composed of a single board computers and located between the edge and the center of the metering infrastructure. In this way,

Standard	Subject
IEC 61850	Communication networks and systems for power utility automation
IEC 61970	Energy management system application program interface including the common information model
IEC 61968	System interfaces for distribution management
IEC 61400-25	Communications for monitoring and control of wind power plants
IEC 62325	Framework for energy market communication
IEC 62351	Standard for the data transfer security
IEC 62056	Data exchange for meter reading, tariff and load control
IEC 61508	Functional safety of electrical/electronic/programmable electronic safety-related systems
IEC 61131	Programmable controllers
IEC 61334	Distribution automation using distribution line carrier systems
ISO/IEC 14543	Home Electronic System (HES) architecture
IEC 61499	Distributed control and automation
IEEE 1547	IEEE Standard for Interconnection and Interoperability of Distributed Energy Resources with Associated Electric Power Systems Interface

Table 2.1 The table reports some of the over 100 IEC standards involved in the Smart Grid communication[4]

they can reduce the amount of data exchanged. However, they do not support edge-computing features nor provide self-configuration and auto-update functionalities.

Ou et al. [40] describe a metering system for transmission networks using wireless technologies. They receive measures from the transmission line and communicate them to a central monitoring service. Unfortunately, their infrastructure does not support third-party applications and the meters are used only to transfer data without performing any data processing.

Lloret et al. [41] introduce a centralized architecture supporting different smart devices by using distinct communication protocols. Their infrastructure includes

Application	Bandwidth	Latency
Substation Automation	9.6-56kbps	15-200 ms
Overhead Transmission Line Monitoring	9.6-56kbps	15-200 ms
HEM	9.6-56kbps	300-2000 ms
AMI	10-100kbps per node, 500kbps for backhaul	2000 ms
Wide-Area Situational Awareness Systems (WASA)	600-56kbps	15-200 ms
Demand Response Management	9.6-56kbps	several minutes
Outage Management	9.6-100kbps	2000 ms
Distribution Automation (DA)	99.0-1500Kbps 99.99%	20-200 ms
Distribution Management	14-100kbps	ms-100 sec
Securitysection	99.0% per node	2000 ms
Distributed Energy Resources and Storage	56kbps	2000 ms
Vehicle to Grid (V2G)	56kbps	300 ms-2 sec
Electrical Vehicles (EVS) Charging	9.6-56kbps	ms-5 min
Asset Management	9.6-56kbps	ms-5 min
Meter Data Management	9.6-56kbps	$<$ 1 minute

Table 2.2 Bandwith and latency requirements of Smart Grid applications[5]

cloud services using big-data techniques to provide different types of services to users and DSO. However, their approach is not scalable enough for large smart grids. With thousands of devices, a centralized infrastructure implies the necessity to transfer and collect a huge amount of information. This process can cause latency, reduce the bandwidth, and increase network costs.

IoT Protocol	QoS	Data Security	Transport Layer	Message Prioritization	Message Pattern	Complexity
AMQP	✓	TLS SSL	TCP	✓	Req-Res Pub-Sub	Low
CoAP	✓	DTLS	UDP	✓	Req-Res Pub-Sub	Low
CORBA	✓	SSL	UDP	x	Req-Res Push-Pull	Medium
DDS	✓	SSL DTLS	TCP UDP	✓	Pub-Sub	High
DPWS	✓	TLS SSL	TCP UDP	✓	Pub-Sub	Medium
MQTT	✓	TLS SSL	TCP	x	Pub-Sub	Low
OPC UA	x	SSL	TCP	x	Req-Res Pub-Sub Push-Pull	High
XMPP	x	TLS	TCP	x	Req-Res Pub-Sub Push-Pull	High
ZeroMQ	✓	TLS	TCP	x	Req-Res Pub-Sub Push-Pull	Medium

Table 2.3 Comparison of the characteristics of the main IoT protocols used in the Smart Grid context[6]

The previous analysis shows that the most promising approaches are the ones adopting fog and edge computing.

With the diffusion of IoT the Cloud computing approach started to show some limitations. Some of the developed IoT solutions require low latency which can be hard to reach in a traditional Cloud computing architecture where the server can be quite far from the IoT device. Moreover, the rising number of connected devices generates a huge amount of data, however communicating all the collected data to the Cloud could be not useful and sometimes even impossible (for example for security or privacy limitations)[42]. The Edge computing paradigm tries to solve these and other challenges by leveraging the diffusion of powerful IoT devices. Indeed nowadays, sensors, actuators, and other IoT devices have a computational power that is sufficient to perform the data processing on the device itself. The name "Edge" derives from the fact that those devices are located in the outmost part of the network as close as possible to their owner/user. However, in some cases this approach is not applicable: the computational power of the IoT devices could be not enough to perform any kind of data processing or even these devices may need to have strict energy requirements and low energy consumption. In those cases the Fog paradigm can be a solution: this approach provides an intermediate layer between the edge and the cloud relieving some of the computational efforts from both sides. This change of paradigm is affecting also the smart grid scenario. Indeed, various literature contributions demonstrated that various applications, in the smart grid context, can improve their performance in terms of scalability, latency, data storage and processing [43][44] [45] [46]. For these reasons in this work, we proposed a platform capable of supporting both edge and fog computing as those are state-of-art approaches that bring numerous benefits to Smart Grid applications.

Unlike the previous works, i.e., [37, 40, 41], where the computation is performed in centralized form, with fog and edge computing the computational load is distributed among different devices. This strategy reduces, especially for critical applications, the latency and the bandwidth required to transfer information [47]. Forcan et al. [48] locate fog servers close to the metering devices. Servers perform data processing whereas the metering devices are just used for sensing. Liu et al. [49] use edge computing to support multiple network management algorithms, reduce the computation time, and decrease the bandwidth required for data transmission. Their infrastructure, as in all other works mentioned above, makes use of IoT to gather data from the Smart Grid. As [37, 41], and [49], they support the execution of different algorithms to improve and simplify smart grid management, and to increase its reliability.

Another key element of state-of-the-art metering infrastructures is the meter architecture. To have distributed, or even fully delocalized algorithms, and to reduce the necessity to communicate a huge amount of data [50], smart meters should be able to communicate through the Internet and to perform data processing. Unfortunately, the majority of the works designing smart meters mainly concentrate on the performance [51–53] of the physical device. On the contrary, there are relatively few works focusing on the development of smart meters using IoT technologies to enable new features for grid management.

Gallano et al. [54] design an IoT meter supporting a mobile application able to display the energy consumption. However, the meter is not included in a metering infrastructure, and it offer the possibility to run neither on-board nor distributed algorithms.

Using a single board computer, Sirojan et al. [55] develop a 3-phase meter able to collect the values of the current on each phase. The meter is also capable of performing simple data analysis, but it does not offer the possibility to run custom algorithms.

Chen et al. [56] test an IoT-enabled single-phase smart meter for demand-side management in smart homes only. Even if the meter is used as an edge computing device, only a few predefined algorithms can be executed on it without any possibility of customization.

Pegoraro et al. [57] illustrate a 3-phase smart meter prototype capable of running distributed state estimation. Their meter adopts IoT-based cloud systems for data exchange. Pignati et al. [58] suggest a similar solution in which the meter exploits the ICT system of a university campus. In both these works, the meters are used to collect information and to transfer it to concentrators located toward the edge of the communication network. These concentrators run the first step of a distributed algorithm performing data processing.

As mentioned before, both interoperability and flexibility need to be taken into account during the development of a smart device or an AMI. Protocols such DLMS [33] only offer the possibility to securely exchange metering data across the network and to update the meter. However, since it is designed for reading consumer data it could be unfeasible or hard to use it by the multiple concurrent applications for smart grid management.

One of the advantages of our approach, with respect to the ones described so far, is the possibility to use the meter to execute stand-alone or distributed applications. In addition, it is possible to completely customize the smart-meter, both in terms of hardware and software. Thus, the user is free to select the desired target accuracy and the applications to deploy on the meter. Moreover, our infrastructure supports self-configuration and auto-update strategies. In this way, our meters can automatically discover newly installed devices and we can change settings and software applications on the fly. As a consequence, we can deploy on the grid at run-time smart devices which differ from the designed meter and offer innovative and previously unforeseen services.

To summarize, Table 2.4 shows the main similarities and differences between our approach and the works previously described.

2.3 Metering infrastructure and smart meter

This section describes our metering framework. It first details our distributed software infrastructure, defining the actors involved and the technologies adopted to enable the communication among them. After that, it presents the design of our internet-connected 3-phase smart meter (the 3SMA) highlighting its main characteristics and its interactions with the other units. Finally, it reports the decentralized

Infrastructure	Meter prototype	Onboard intelligence		IoT	Computing			Self configuration	Auto update	Third party services
		Data processing	Custom algorithm		on Cloud	on Fog	on Edge			
Proposed	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Gallano et al. [54]	✓	-	-	✓	-	-	-	-	-	-
Sirojan et al. [55]	✓	✓	-	✓	-	-	-	-	-	-
Chen et al. [56]	✓	✓	-	✓	-	-	✓	-	-	-
Pegoraro et al[57]	✓	-	-	✓	-	-	-	-	-	-
Pignati et al. [58]	✓	✓	-	✓	✓	-	-	-	-	-
A. Meloni et al. [37]	-	-	-	✓	-	-	-	-	-	✓
Q. Ou et al. [40]	-	-	-	✓	-	-	-	-	-	-
Chen et al. [38]	-	-	-	✓	-	✓	-	-	-	-
Yan et al. [39]	-	-	-	✓	-	✓	-	-	-	-
J. Lloretet al. [41]	-	-	-	✓	✓	-	-	-	-	✓
M. Forcan et al. [48]	-	-	-	✓	✓	✓	-	-	-	-
Y. Liu et al. [49]	-	-	-	✓	✓	-	✓	-	-	✓

Table 2.4 A synoptic table comparing our approach with several state-of-the-art similar works. The table considers both the infrastructure architecture and the 3SMA design.

applications used to test both the 3SMA and the infrastructure. We focus on auto-reconfiguration for self-healing and state estimation capabilities. Notice that the system supports both on-board (deployed and executed directly on the 3SMA) and remote (deployed and executed on-fog or on-cloud) applications. Moreover, note that the algorithms presented in this work are just a mere example of the ones that could be used on our infrastructure. Indeed, thanks to the auto-update feature, both the infrastructure and the 3SMA meter have been designed to be flexible in terms of updating, adding, and replacing the application at run-time, without the necessity to reboot the whole system. Thus, our infrastructure will also trigger the possibility to use other innovative services in the future. Various tools and programming languages have been used to develop the infrastructure and its actors, in particular: the Device Catalog, the Network Reconfigurator and the code for the data collection on the meter were all developed using Python 3.6 The algorithm for the Faull Location and the one for the State Estimation was developed in Matlab. The Device Catalog and the Network Reconfigurator were executed on a desktop PC through a Docker container. The same goes for the two MQTT brokers which used the Mosquitto Docker image. The algorithms running on the 3SMA, instead, were executed through Octave as in most cases (such as the ones we tested) is a drop-in replacement for Matlab.

2.3.1 Distributed Metering Infrastructure

One of our targets is to design a highly scalable, distributed, and decentralized metering infrastructure to manage and monitor a high number of devices and services over the grid. As shown in Figure 2.1, the principal entities of this infrastructure are the *Message brokers*, the *Device Catalog*, the *3-phase Smart meter* (3SMA), and the *Remote service* applications. These units talk to each other by exploiting two communication paradigms, namely the request/response through the REpresentational State Transfer (REST) Web Services [59] and the publish/subscribe [60] through the Message Queuing Telemetry Transport (MQTT) protocol [61].

The Device Catalog (DeC), at the center of Figure 2.1, is an essential software component of our infrastructure as it acts as both service and device register. It is in charge of registering all active devices and software entities managing the smart grid. It also keeps track of the active units connected to the Internet, thus enabling service and device discovery. When, for any reason, a hardware device or a software entity cannot communicate with other units, the DeC marks it as unavailable and propagates this information to all other units connected to the infrastructure. This procedure makes the unavailability of a device known on the entire network. The DeC also stores the metadata for each 3SMA. Metadata includes the algorithms run by the meter and the topology of the portion of the grid that it has to monitor. The DeC can eventually provide this information to the remote service applications, such that these services are able to perform the right set-up to retrieve the required data. The DeC is also responsible for the self-configuration and auto-update of the 3SMA units. In other words, the DeC provides to each meter the endpoints of the message brokers, the metadata, and all the necessary information to keep the meter updated. Moreover, it provides to each remote service application a complete list of both 3SMA and other services' endpoints, and all the necessary information to manage the smart grid. As mentioned before, two communication protocols, REST and MQTT, are used in this framework. REST is a synchronous architectural style used to build web services exploiting the HyperText Transfer Protocol (HTTP) protocol [59]. It is widely supported and used for 1-to-1 communications. On the contrary, MQTT is an asynchronous protocol implementing the publish/subscribe paradigm [60]. It is quite common in the IoT world and it is optimal for 1-to-many and event-driven communications [61]. Due to their differences and their specific characteristics, these two protocols have been selected to perform different tasks. REST is mainly used for the self-configuration and auto-update of the IoT devices deployed over the grid. MQTT is adopted to exchange data with the meters and on remote service applications. More details on these protocols are reported in Section 2.4.

The two message brokers of Figure 2.1 manage the publish/subscribe communication among hardware and/or software components exploiting the MQTT protocol. We decided to use two different message brokers to separate the two main communication contexts. The first broker manages data exchange over the grid (i.e., topology changes, reconfigurations, etc.). The second broker monitors information. We adopt this configuration to minimize the delay of the MQTT messages related to device management. Indeed, depending on the amount of information we need to exchange and on the implementation of the message broker, a single broker might constitute a bottleneck for our infrastructure, thus increasing the overall communication latency. As communication latency is a crucial factor in smart-grid management, we appropriately considered it and we minimized it by resorting to two message brokers in our configuration.

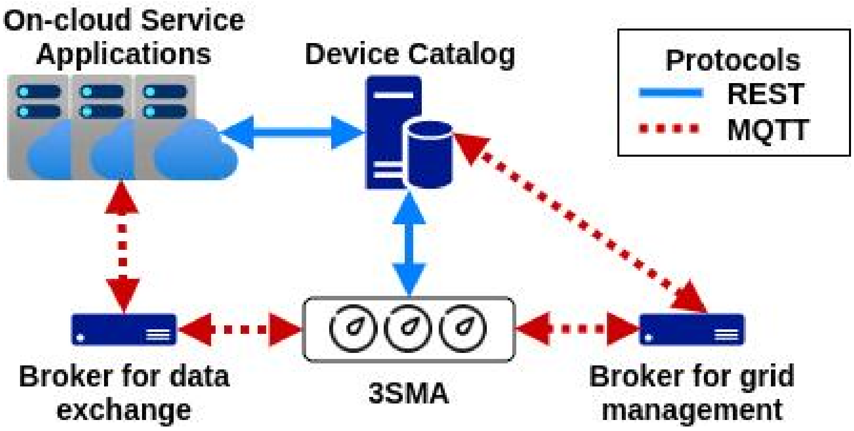


Fig. 2.1 Infrastructure schema.

2.3.2 Our 3-phase Smart Meter Architecture

We designed our 3SMA to operate in our distributed infrastructure, described in Section 2.3.1, and to fully exploit its features. Essentially, we optimized our 3SMA to perform three different tasks: self-configure, collect electric measures from the power grid, and run user-defined network algorithms. These tasks are logically performed by different units, as graphically illustrated in Figure 2.2. The tasks performed by each unit are the following.

The first unit, i.e., the communication interface, is represented on the left-hand side of Figure 2.2 and it is in charge of enabling the communication over the Internet among 3SMA and the other actors in the platform, either hardware or software.To be compliant with the infrastructure, we exploit both the communication protocols introduced in Section 2.3.1, namely REST and MQTT. We use REST, i.e., the request/response approach, to gather the initial configuration of the 3SMA from the DeC unit. We also use the REST protocol to notify the DeC device that it is reachable and it is properly connected to the network. Finally, we use REST to update the 3SMA in terms of the endpoints, the MQTT broker that must be used, and the algorithms that must be run. On the other hand, we adopt the MQTT protocol, i.e., the publish/subscribe approach, to receive updates from the DeC unit concerning the status of the grid and the settings of the metering infrastructure (such as modifications on the topology of the grid, changes of the endpoint, etc.). Moreover, we also use the MQTT protocol to transfer the data collected by the 3SMA unit and the results computed by the on-board algorithms to the remote service applications.

The second logic unit of our 3SMA device, i.e., the self-configuration and algorithm unit at the center of Figure 2.2, is in charge of executing three main different tasks: Self-configuration, auto-updating, and data-processing. We stress the terms "self" and "auto" because one of the main features of our meter is that it can potentially run any algorithm, at least as far as it has the computational power to execute it. When we need to execute a new algorithm on the meter, we start by updating the algorithm inside the DeC device. Once done this, the 3SMA will receive all updates from the DeC, it will retrieve the new algorithm, and it will start executing it. This strategy can then be used to replace any existing algorithm on-the-fly, without affecting the rest of the system, and enabling our infrastructure to be ready for future extensions.

The third, and last, logic function performed by our 3SMA, i.e., the data acquisition part, is performed by the rightmost block of Figure 2.2. This unit collects all electrical quantity measures on the power grid and it makes them available for the algorithmic manipulation performed by the computational unit.

Obviously, the three logic units previously described can be realized, i.e., implemented, adopting different hardware devices. Our current hardware prototype combines a Raspberry Pi 3 Model B unit with a Data AcQuisition board (DAQ) provided by Measurements Computing™ [62]. The Raspberry Pi unit is a small embedded computer and it is connected with the DAQAQ using a USB port. We selected these hardware devices because using open-source and relatively cheap units drastically reduces the costs of the entire infrastructure and improves its adoption in large networks improving our design scalability. As mentioned before, indeed scalability is one of the main challenges faced by modern infrastructures. With this hardware configuration, our DAQAQ is able to collect six channels, each one with a frequency that is limited to 6.4 kHz but it can be customized according to the necessity of our application. For example, for our initial tests, we limited the sampling rate to 3.2 kHz, i.e., to 50% of the maximum possible frequency. Anyway, this value was not only more than sufficient to perform our acquisition phase but it also allowed us to dedicate the remaining computational power to run our algorithms more efficiently.

Using this configuration, the measures are collected through the DAQAQ by the Raspberry Pi. The Raspberry Pi is also in charge of running the self-configure and auto-update routines, of executing the different algorithms, and of exchanging information with the remote service applications and the DeC. Thus, it is essentially the hardware unit making our 3SMA architecture “smart”.

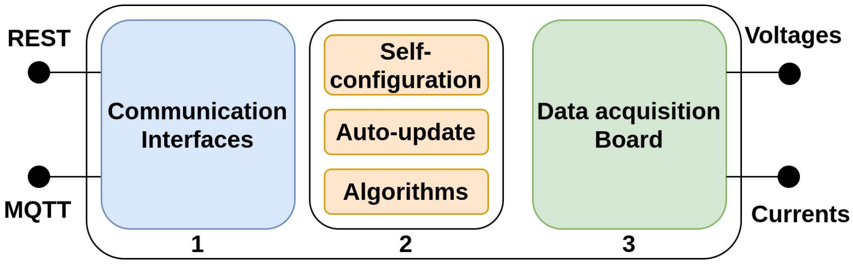


Fig. 2.2 Conceptual scheme of the 3SMA on-board software.

2.3.3 Algorithm and remote service applications

As mentioned in Section 2.3.2, our 3SMA supports on-board applications deployed and executed directly on the device. In general, these applications can be seen as a part of a more complex and complete software suite running in a distributed way over the network, thus offering on-edge, on-fog, and on-cloud features. To evaluate the computational power of our infrastructure, we tested it with two different and well-known applications: a FLISR algorithm and a SE one.

Given a network fault, a FLISR procedure automatizes the localization and the isolation of the portion of the grid affected by the fault and, wherever possible, it reconfigures the power grid to minimize the number of disconnected customers [63]. This is a combination of fault location algorithms with Fault Detection Isolation and Restoration (fdir) schemes to improve the performance of fault management. The aim is to restore most of the interrupted customers as quickly as possible.

To reach these targets, we deployed the algorithm proposed by Estebsari et al [63] on our 3SMA (i.e., on-edge). Their approach combines an impedance-based algorithm with a sparse voltage measurement-based strategy. Their procedure detects the bus pair affected by the fault and evaluates the distance between the fault and the first bus of the pair. The bus pair defines the secondary substations (i.e. bus) and the branch on which the fault occurred. When a fault occurs, the protection relays at the beginning of the feeders trip, and the circuit breakers disconnect the feeder from the supply. The required measurements to run the above-mentioned fault location algorithm are sampled by 3SMAs and sent to the one with the activated algorithm and installed at the beginning of the feeder. Once the measurements are received by this 3SMA, it runs the fault location algorithm. The results of this computation are then sent to the remote service application running on the cloud. This application reconfigures the network, providing a new grid topology to isolate the faulty portion of the network and to restore the power supply for the rest of the grid. To obtain the new network configuration, the procedure sends an actuation message to open or close every smart switch that needs to be switched. The new topology is also communicated to all entities (both hardware and software) that can be affected by the changes. Additional details on the communication phase of the FLISR process are given in Section 2.4.2.

The main target of a SE procedure is to provide an estimation, over the entire network, of some electrical values (such as voltages, powers, and currents), given the available local measures. This task is of paramount importance to promptly identify any critical problem in the grid, especially when distributed or renewable energy sources are connected to the system, as the energy production of those sources is largely unpredictable and volatile. Usually, system operators estimate the state of the network by adopting a centralized application collecting all the measures coming from the meters spread across the grid. Unfortunately, this process can be unfeasible on smart grids, or at least it can be very expensive, due to communication bottlenecks and unpredictable delays. Therefore, to distribute the computation on-edge and on-fog, we adopted the hierarchically and distributed SE algorithm introduced by Pau et al. in [64, 65]. This algorithm is based on the weighted least squares approach. Its target is to filter out the errors on the measures to provide the most probable operational state of the grid, exploiting the redundancy of the input measures (please, see [65] for more mathematical details on the algorithm). The distributed implementation of the algorithm divides the estimation problem into multiple hierarchical SE levels, which are sequentially executed following a bottom-up process. In our implementation, the estimation process is divided into three different levels: The *Concentrator*, *Low Voltage (LV)*, and *Medium Voltage (MV)* level. This partition reflects the hierarchy existing on the grid and illustrated in Figure 2.3. In this representation, multiple concentrators *C* are connected to a single *LV substation*, and multiple LV substations are connected to a single *MV substation*. These connections can be easily represented with a tree structure which can also be exploited for the logical partition of the SE process. The first computation is made at the concentrator level. At this level, the state is estimated for all customer nodes subtended by the feeder bus to which the concentrator is associated. The LV level estimators use the results provided by the concentrators (voltage and total power at the feeder bus) to estimate the operating conditions of the low voltage grid feeders. Similarly, the MV level estimator uses the result of the MV and LV level (i.e., voltage and overall power) as input to evaluate the operating conditions of the subtended medium voltage grid. Differently from the conventional implementation of the SE algorithm, in our architecture both the LV level and the MV level state estimation algorithms are deployed on a 3SMA entity. Each 3SMA, in charge of the execution of the SE service, executes one of the two algorithms (namely, either the LV level or the MV level) according to the settings that it has received during the self-configuration and the auto-update procedure. These are two on-fog algorithms. More details on the communication flow and the coordination among the different actors running the distributed SE are provided in Section 2.4.

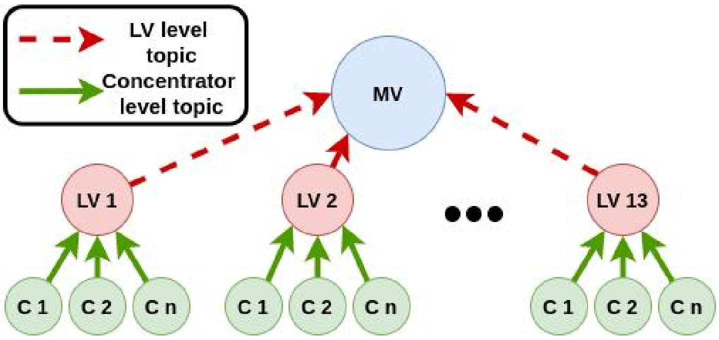


Fig. 2.3 Conceptual scheme of the distributed State Estimation algorithm.

2.4 Communications flow

As analyzed in Section 2.3, we adopt both the requests/response and the publish/subscribe communication paradigms [60]. We use REST during the initial communication phase. In this step, we define the settings of the DeC and 3SMA units, and we respect the client-server approach required by the protocol. On the contrary, we adopt the MQTT protocol every time an actor of the infrastructure needs to transfer information, even in (near-) real-time, to more than one unit. In this case, the publish/subscribe functionalities offered by MQTT completely satisfy multi-point communication requirements. Given this framework, this section describes the communication scheme used between the units of our infrastructure. More specifically, we detail how REST has been used for the self-configuration and auto-update phases, and how the MQTT protocol has been exploited to manage faults (i.e., the FLISR service) and to coordinate the state estimation process.

2.4.1 Self-configuration and auto-update

Figure 2.4 illustrates the main steps of the self-configuration process, which is also the first phase running within the infrastructure every time a new 3SMA is installed or (re-) booted on the grid. The self-configuration capability avoids the necessity to manually configure each 3SMA and it also enables remote reconfigurations. These possibilities increase the scalability and maintainability of the whole infrastructure. As stated in Section 2.3, the DeC stores all settings required by each 3SMA. As soon as a 3SMA is deployed and turned on, it sends a configuration request to the DeC, specifying its unique (device) identifier. Once this request is received, the DeC performs two tasks. First, it adds the specific device to its list of active units. Then, it sends back to the 3SMA the full list of settings. When the 3SMA receives its configuration setting back, it initializes its status following the procedure specified by the settings. Immediately after that, it starts collecting and sending data, running the specified algorithms and it publishes and/or subscribes to the MQTT topics indicated in the settings.

The other major feature of the 3SMA is its auto-update capability. Auto-update is illustrated in the bottom frame of Figure 2.4. The first step of this phase is similar to the self-configuration process. However, in the auto-update phase, the 3SMA periodically inquires the DeC to receive its settings and the DeC replies by updating its configuration. In addition, every time it receives a request from a 3SMA, the DeC stores the time-stamp of such a request, which is needed to keep the list of active and online devices up-to-date. Indeed, if the DeC does not receive an auto-update request from a specific 3SMA, after a certain amount of time (configurable by the user), it marks the 3SMA as "disconnected" until the next auto-update request. It is worth noticing that the self-configuration and the auto-update features increase the flexibility and the configuration and re-configurations speed of the whole system.

For the purpose of the testing scenario, the parameters used for the self-configuration of the meters were the following

- The topology of the monitored portion of the grid

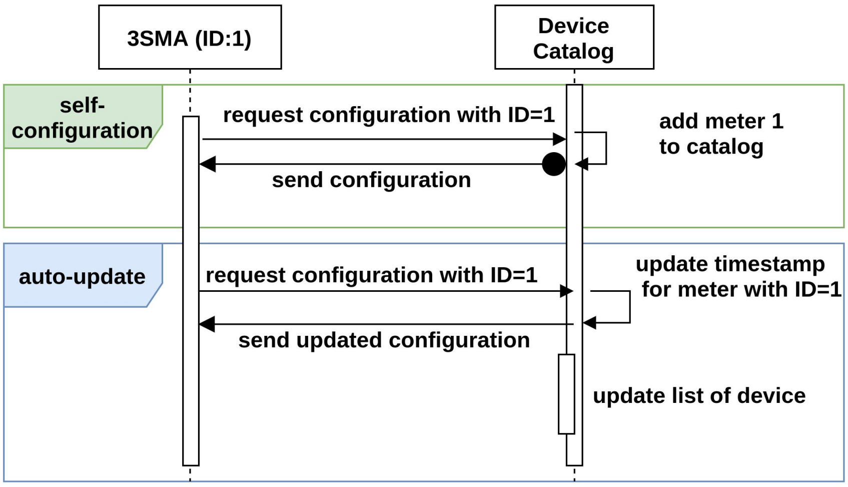

3SMA (ID:1)
Device Catalog
self-configuration
request configuration with ID=1
add meter 1 to catalog
send configuration
auto-update
request configuration with ID=1
update timestamp for meter with ID=1
send updated configuration
update list of device

Fig. 2.4 Self-configuration and the auto-update: The main operational steps of our infrastructure.

- The algorithm deployed on the meter and whether they were enabled or not
- The MQTT topic at which publish and/or subscribe

An example of a JSON configuration used is the following:

```
{
        "meterID": 000000,
        "topology": {...},
    "DeCIP":"10.0.0.10",
        "algorithms": [{
                "name": "FaultDetection",
                "enabled": true
        }, {
                "name": "StateEstimation",
                "enabled": false
        }...],
        "MQTT": [{
                "app": "RMS",
                "topics": {
                        "voltage": ["3SMA/data/voltage/00000"]
                }
        }, ...],
}
```

An example of the topology field is the following:

```
{
        "nodes": [{
                        "id": "1",
                        "label": "Node 1"
                },
                {
                        "id": "2",
                        "label": "node B"
                },
                {
                        "id": "3",
                        "label": "node C"
                }
        ],
        "links": [{
                        "source": "1",
                        "target": "2"

                },
                {
                        "source": "1",
                        "target": "3"
                },
                {
                        "source": "2",
                        "target": "3"
                }
        ]
}
```

The configuration of a device therefore is determined by its "meterID" field. Once a meter connects to the catalog the configuration corresponding to its "meterID" is communicated. The only manual step involved in the initial configuration is the upload of a file, similar to the one in the example, on the Device Catalog, any further modifications to the configuration are automatically retrieved through the auto-update mechanism.

2.4.2 Fault Location Isolation and Service Restoration

As illustrated in Figure 2.5, when a fault is detected, the 3SMA seeks the bus pair on which the fault occurred. Once the search is finished, the identifiers of these 2 buses, and the distance of the fault from the first bus of the pair, are published using the MQTT protocol. At this point, the remote service application in charge of the network reconfiguration, which is subscribed to the MQTT protocol, receives these identifiers and computes the new configuration of the network. While doing that, this application also takes into account the detected fault, so that the line between the two faulted buses is not included. Once this step is performed, the network configurator sends a command to the switches that need to open or close the connection to reconfigure the grid. In our tests, all open switches that could receive this kind of actuation commands were simulated on the real-time simulator. At the same time, the information related to the new topology of the grid is transferred to the DeC. When the DeC receives the updated topology, it uses MQTT to store and forward this information to all 3SMA units that are affected by the configuration change, such that these 3SMAs can update their settings accordingly. If, for any reason, a 3SMA device does not receive such update settings in (near-) real-time, it will receive them when it runs its auto-update routine (please, see Section 2.4.1). Notice that updating the setting is particularly important for the correct execution of the SE. When a network reconfiguration occurs, a 3SMA may need to run the algorithm for a different portion of the grid. In this case, the new settings are needed to reconfigure both the SE algorithm and the MQTT communication protocol.

2.4.3 State estimation

As mentioned in the section above, our distributed SE algorithm includes three layers: The concentrator-level, the LV grid-level, and the MV grid-level. Figure 2.6 describes how the MQTT protocol communicates results between different layers. Each concentrator-level SE is an MQTT publisher and uses the data coming from the downstream of an LV grid node as input data. Once the computation is finished,

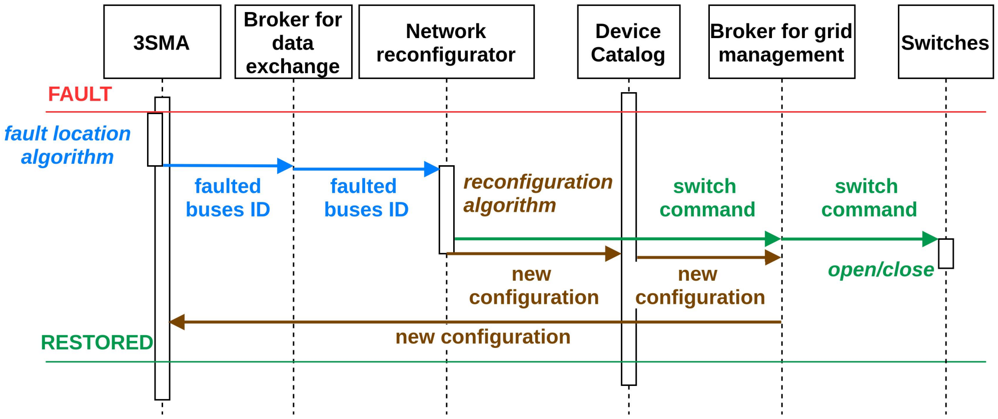


Fig. 2.5 Fault detection and restoration process: Communication flow among the different units of the grid.

the results are published through the MQTT protocol with a topic that identifies the associated LV node. The LV grid-level SE algorithm is subscribed to the MQTT topics that identify each concentrator of the corresponding portion of the grid to receive their SE results. These results are used as input to evaluate the status of the low voltage grid. After the previous evaluation, the new results produced by each LV state estimation are published using the MQTT protocol that identifies the corresponding MV portion of the grid. Therefore, they are received by the MV grid-level SE. In turn, this level executes the last part of the distributed algorithm. The overall results are then published, using, again, the MQTT protocol. Each potentially interested remote service for grid management can obtain this information by subscribing to the related topic.

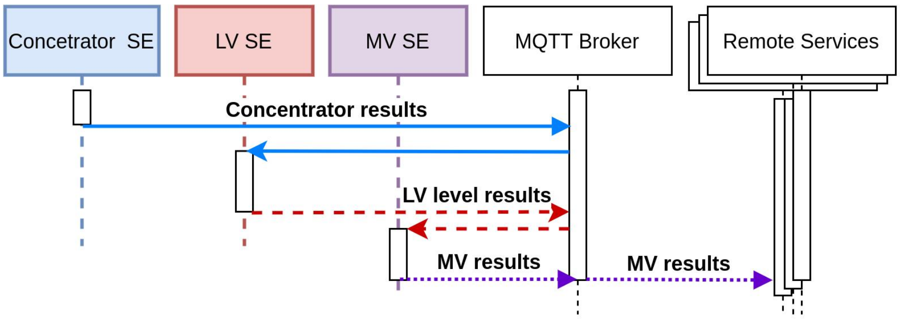


Fig. 2.6 Estimation process: Communication flow among the different layers and units of the grid.

2.5 Case Study and Experimental Setup

In this section, we describe our testing methodology for the 3SMA unit and the distributed metering infrastructure defined in Section 2.3. To perform the verification phase, we exploit the multi-model co-simulation platform introduced by Barbierato et al. [66]. This platform provides an environment to realistically simulate different smart grid scenarios. We focus on the simulation performed with Hardware- and Software-in-the-Loop.

To perform our simulations on different configuration scenarios, we model a small portion of an urban distribution grid with two feeders and one normally open switch. With this network, we simulate a small portion of the real distribution grid of Turin, a large city in the northwest district of Italy. Figure 2.7 represents the topology of the case study and the location of the fault. Overall, the grid is composed of two primary and twelve secondary substations, with a high voltage of 220 KV and a medium voltage of 22 KV. All twelve secondary nodes supply residential LV grids. In our tests, we also consider a constraint on the radial operation of the grid. It is important to notice that the switches are modeled as IoT devices simulated on the Opal-RT. Once the fault is isolated, we use this framework to send the necessary actuation messages (adopting the MQTT protocol) to all involved switches to restore the power supply.

To run the simulations of the power network, we use the Opal-RT$^{©}$ Digital Real Time Simulator. The model of the grid, implemented in MATLAB$^{©}$ Simulink using the RTLab$^{©}$ software provided by Opal, can be compiled and uploaded in the simulator. The simulator is able to execute the simulation with a fixed time step of 250 μs. During the simulation, the Opal-RT can provide up to 16 analog outputs. During our tests, we also simulate all sensing devices that are required on a real-world network. The output signals of the Opal-RT are scaled in the range $[0, +10]$ V to respect the input range of the DAQ. Thus, the DAQ is directly connected to Opal-RT via the analog outputs. Figure 2.7 shows the topology of the grid implemented in the simulation, the location of the 3SMA units, and the location of the simulated devices (both smart meters and switches). To perform our test with Hardware-in-the-Loop, we substitute 2 out of the 14 smart meters by our real 3SMA prototypes. On the one hand, our 3SMAs are connected to the Opal-RT with their DAQs to sense and collect power measurements from a primary and a secondary substation (respectively, M1 and M2, in Figure 2.7). On the other hand, they are connected to the Internet to be automatically configured and to periodically send information (as described in Section 2.4). To perform more realistic simulations, the simulated devices (i.e., the remaining smart meters and switches) send and receive information via the Ethernet connection of the Opal-RT. Among the simulated devices, we also include those deployed at the low voltage level running the state estimation.

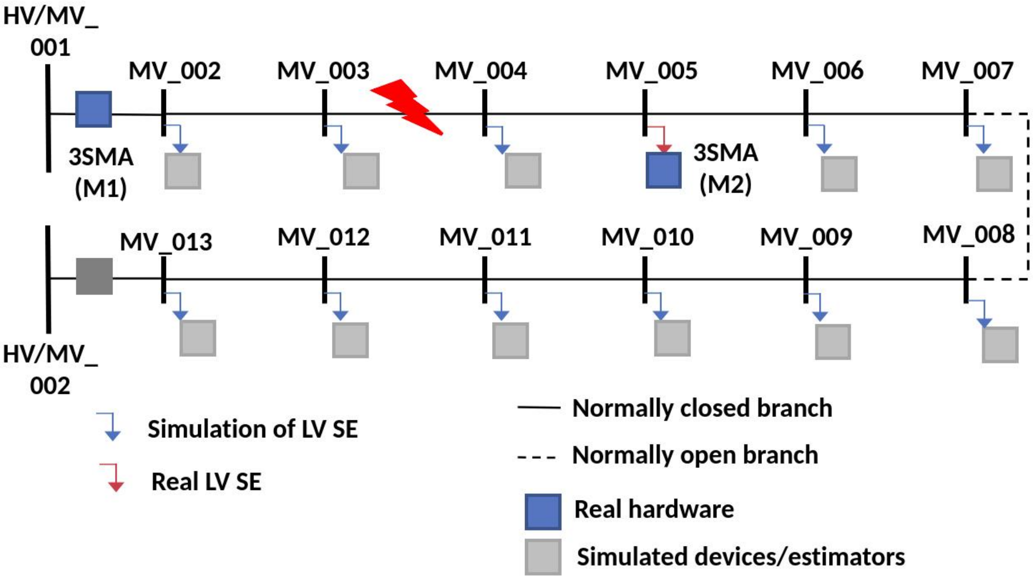


Fig. 2.7 Our simulated grid: Its topology with the location of all 3SMA prototypes and all other simulated devices.

As described in Section 2.2, one of the crucial aspects for any service implemented on a smart grid is the communication latency required to fulfill a request (measured as the time passed from the moment in which the event has been discovered and the final actuation). To take into account the Internet congestion during our evaluation, we use some of the features provided by our multi-model co-simulation platform [66]. Indeed, this platform integrates different network simulators to realistically simulate a MAN. For our simulations, we choose Mininet [67]. Figure 2.8 reports the schema of our MAN backbone model. Nowadays, MAN backbones typically leverage upon fiber-optic links deployed across our cities that interconnect different backbone routers forming a ring configuration. Fiber optic links usually guarantee 100 Mbps connections and, in our model, they are full-duplex with a maximum length of about 10 Km and zero losses (Figure 2.8 represents these links with red edges). In Figure 2.8, we supposed that all smart devices, simulated in

Opal-RT, and the two 3SMAs prototypes are connected to the same backbone router R1.

The remote service for Network Reconfiguration, the DeC, and the two MQTT message brokers, are connected to routers R2, R3, and R4, respectively. R1, R2, R3, and R4 are connected to their respective sub-networks through a 10 Mbps full-duplex link, with a maximum length of about 1 Km, and zero losses (Figure 2.8 represents these links with green edges). Finally, routers R5, R6, and R7 serve three other different sub-networks that generate background traffic with different rates to realistically congest the MAN by exploiting iPerf [68]. To avoid bottlenecks, the links between these last three routers and their traffic generators are 200 Mbps full-duplex, with a maximum length of about 1 Km, and zero losses (Figure 2.8 represents these links with black edges). Thus in our simulations, the 3SMAs, Opal-RT, DeC, network reconfiguration service, and message brokers are in different locations in the same metropolitan area and communicate over the internet.

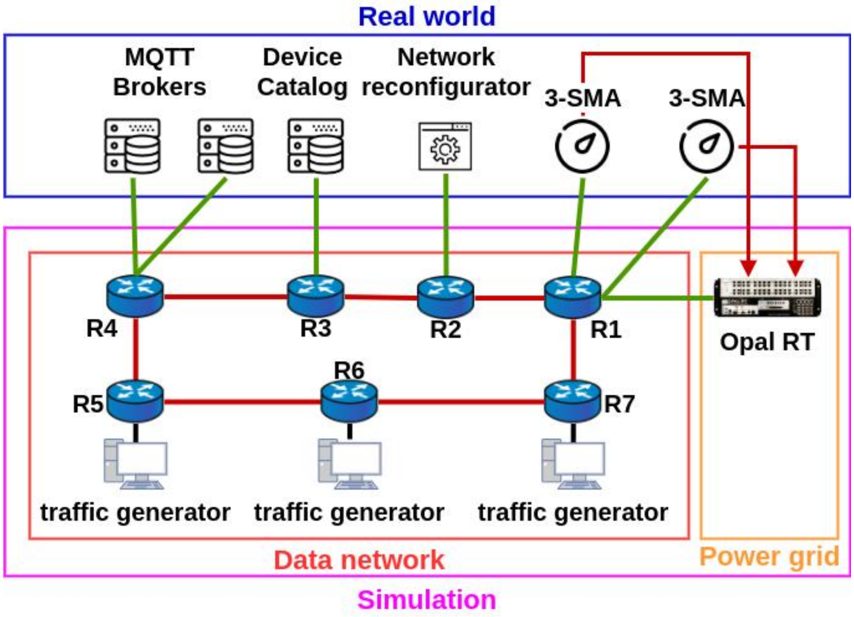


Fig. 2.8 Our MAN backbone model.

Using this framework, we have been able to run a test for two scenarios: One including normal operations and one with some fault occurrences. In the first case, each 3SMA collects the data coming from the analog output of the Opal-RT, namely the 3 voltage and the 3 current values of the 3-phase signal. The data collected are continuously analyzed to check whether a fault has occurred and the state estimation algorithm is executed every minute. In the second scenario, whenever a fault (triggered using the graphical user interface of the real-time simulator) is identified by a 3SMA, the FLISR routine is automatically started to find the location of the fault, isolate it, and restore the service.

In our laboratory set-up, we employ, together with an Opal-RT and the two 3SMA prototypes, four different desktop computers to run the network reconfiguration services, the DeC, the two MQTT message brokers (we choose the Eclipse Mosquitto [69]), and the Mininet Network Simulator. It is worth noting that we implemented two different versions of our 3SMA. The first one follows the hardware specifications discussed in Section 2.3.2, i.e., it uses a Raspberry Pi 3 Model B connected via USB with the DAQ provided by Measurements Computing™ [62]. Hereinafter, we refer to this 3SMA as 3SMA-RPi. The second version of 3SMA is used to assess the performance of the first one, and we implement it with a Laptop computer connected with a more precise (and expensive) data acquisition board, namely the BNC-2120[70] and the DAQCard-6062[71]. These boards support a sampling frequency up to 500 KS/s per channel, with smaller input noise, and a quite higher resolution. Hereinafter, we refer to this 3SMA as 3SMA-Laptop.

All these devices belong to the same dedicated local area network. They are physically connected through a 100 Gbps Ethernet switch that introduces a negligible communication latency of about 10 ns. Gigabit Ethernet equipment provides backward compatibility to older 100 Mbps and 10 Mbps legacy Ethernet devices [72]. All the traffic generated by this equipment is routed through the computer where the virtual MAN runs in Mininet. Tables 2.5 and 2.6 reports all the parameters required to our model to configure the simulation scenario in terms of both power grid and MAN.

Node type	# of Node	Voltage	Distance among nodes
HV to MV substation	2	220kV/22kV	n.d.
MV to LV substation	12	22kV/230V	220 m

Table 2.5 Electric parameters used by our model to set-up the simulation.

Node type	Connection type	Transfer rate
Backbone routers connection	full-duplex,max 10 Km,zero losses	up to 100 Mbp
R1,R2,R3,R4 towards users	full-duplex,max 1 Km, zero losses	up to 10 Mbp
R5,R6,R7 towards traffic generators	full-duplex,max 1 Km, zero losses	up to 200 Mbp
Physical connection	Gigabit Ethernet switch	up to 100 Mbp

Table 2.6 Our simulated grid: The main parameters used by our model to set-up the simulation.

2.6 Experimental Results

In experimental analysis, we consider two sets of experiments. The first one includes evidence on fault location isolation and service restoration (FLISR). The second one reports data on state estimation (SE). Each aspect is analyzed in one of the following subsections.

2.6.1 Fault Location Isolation and Service Restoration

In this section, we focus on the efficiency of our 3SMA-RPi unit to implement the FLISR procedure. We evaluate the computation efficiency and the accuracy of our 3SMA-RPi architecture and we also compare it with its laptop implementation, i.e., the 3SMA-Laptop, exploiting a more precise data acquisition board (see Section 2.5). As illustrated in Figure 2.5, one of our targets is to minimize the time required to reconfigure the network when a fault occurs. In a real-world scenario, a crucial factor in measuring this time is the congestion of the MAN, which can strongly influence the communication delays. To take this aspect into account, we consider in our scenario the data network simulation previously described. We also tested different scenarios by changing the location of the fault in the grid to evaluate how this aspect influences our results.

During our tests, the 3SMA-RPi was always able to correctly detect the presence of a fault on the grid, with no false positive or false negative. Moreover, the fault location algorithm was always able to correctly identify the two buses among which the fault occurred and locate the fault with an error of ±50 m. Notice that this relatively low accuracy is strongly related to the quality of the USB DAQ used to collect the values of voltage and current. To double-check this accuracy, we repeated our test replacing the 3SMA-RPi with the 3SMA-Laptop. With the new unit, the error to locate the position of the fault was reduced to a value of ±10 m. However, even if this accuracy analysis confirms our expectations, it is worth noticing that to correctly complete the FLISR process the meter just needs to correctly identify the two buses among which the fault occurred. For example, in Figure 2.9 the position of the fault does not have any meaningful effect on the execution time. This was equal to 8.17±0.14 s for the 3SMA-RPi and to 0.531±0.008 s on average using the 3SMA-Laptop.

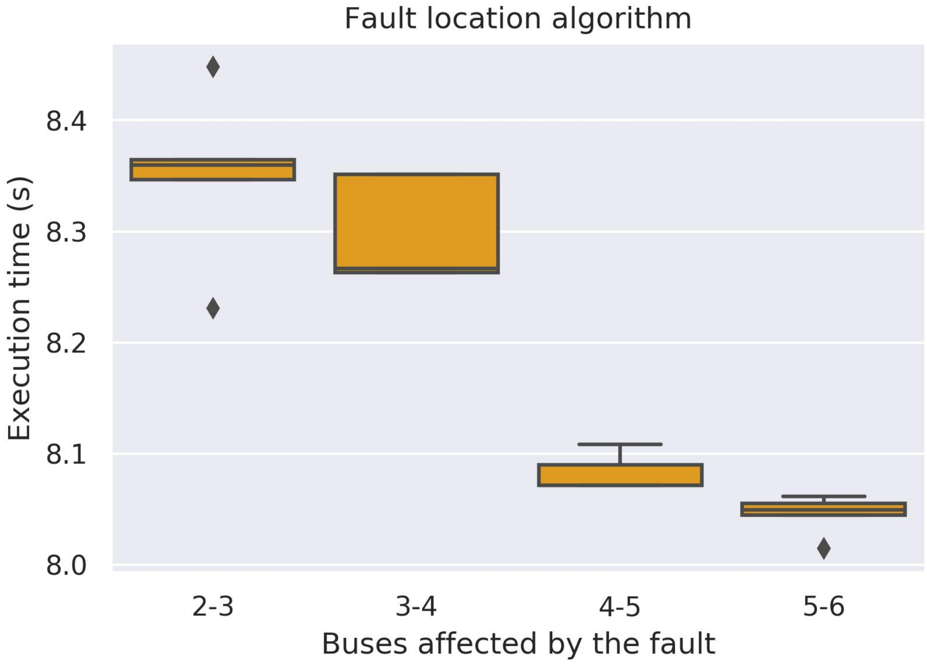


Fig. 2.9 The fault location phase: Execution times exploiting the 3SMA-RPi unit.

To properly evaluate our approach to fault restoration, the last factor we need to take into account is the communication delay. To do that, we measure the elapsed time between the moment in which our smart meter sends the MQTT message containing the information of the buses affected by the fault and the moment it receives the new topology from the network configurator through another MQTT message. The latency time is shown in Figure 2.10, with different levels of network congestion and a packet size of 1.4 KByte. As expected, the latency in the communication is mostly related to the congestion level. Even if this measure includes the execution time of the Network Reconfiguration algorithm, this time does not particularly affect the delay due to the transmission time as it is always lower than 0.028s. In the worst scenario of our simulation, with 100% of congestion, the medium latency time was equal to 0.49 s and the maximum latency time was 1 s. However, Internet service providers try to avoid such high network congestion as, in long periods, it could lead to the collapse of MAN itself. Thus, we can assume that the communication latency would be always smaller than 1 s.

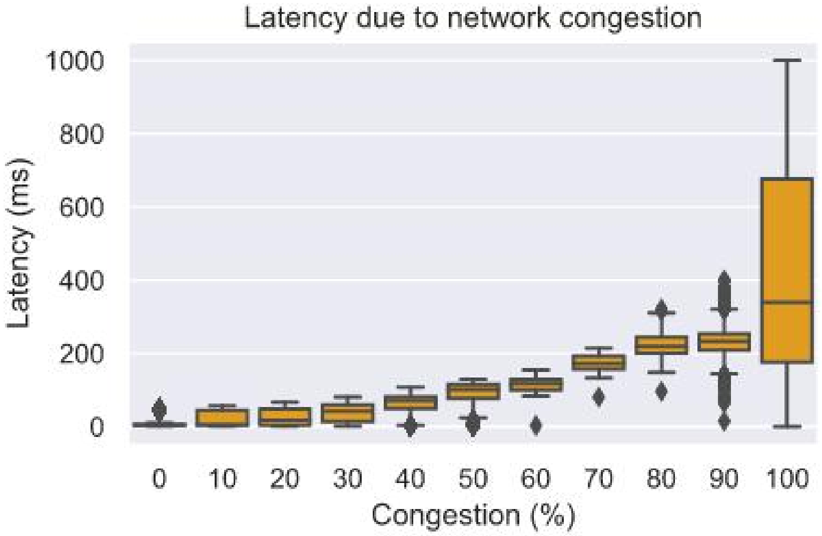


Fig. 2.10 Latency time as a function of the level of traffic over the network. The packet size is equal to 1.4 KByte,

To further detail our analysis, Figure 2.11 reports the time required by all main steps of our procedure in the worst-case scenario, i.e., their higher values obtained running the procedure several times. From the graphic, it is possible to notice that during our simulations the maximum time needed to perform the whole FLISR is about 10.5 s for the 3SMA-RPi unit. This maximum time decreases to approximately 2.5 s for the 3SMA-Laptop device. As shown in Figure 2.11, the majority of this time is taken by the execution of the fault location algorithm running on-board of our smart meter. On the contrary, the latency due to the data communication and execution of the Network Reconfigurator represents a minor contribution. Interestingly, this result also highlights that, with a relatively simple infrastructure and low-cost hardware devices, it is possible to drastically improve the time required by the operator to restore the energy distribution after a fault.

As a final consideration, Figures 2.11 and 2.12 show that the advantage of our approach in terms of service restoration. With the methodology traditionally used in real scenarios, the restoration phase can require from 40 minutes to more than 80 minutes [1] (Figure 2.12). With our infrastructure, the same process is completely automated and, as illustrated by Figure 2.11, just a few seconds are required for the entire process even in the worst-case scenario. Thus, the difference of about 8 s in the computation time between the 3SMA-RPi and the 3SMA-Laptop unit can be considered as negligible. As a consequence, our approach not only improves the quality of the service delivered to the customers, but it also helps the system operators to reduce maintenance costs and it allows them to avoid penalties for long service failures.

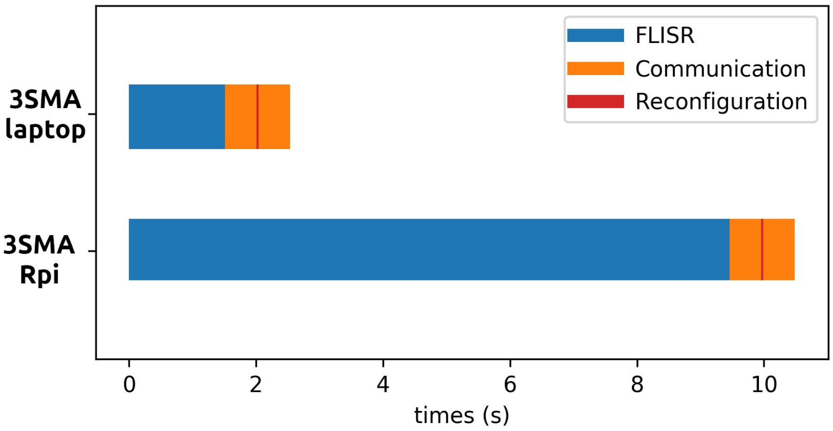


Fig. 2.11 The total FLISR time for the proposed infrastructure: A comparison between the low cost 3SMA-RPi unit and the more powerful but expensive 3SMA-Laptop.

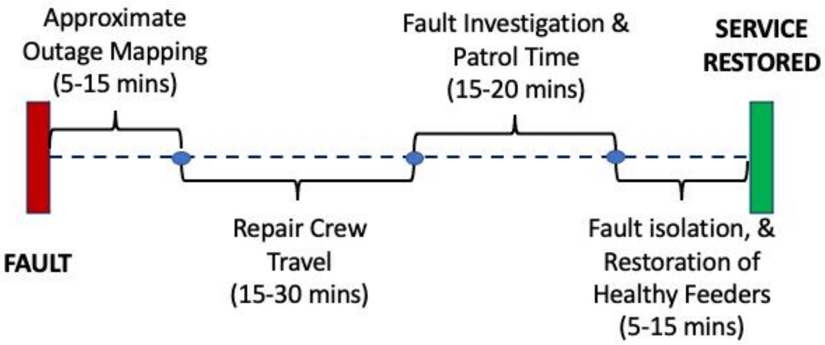

Approximate Outage Mapping (5-15 mins)
Fault Investigation & Patrol Time (15-20 mins)
SERVICE RESTORED
FAULT
Repair Crew Travel (15-30 mins)
Fault isolation, & Restoration of Healthy Feeders (5-15 mins)

Fig. 2.12 Legacy process for fault management on real infrastructures [1].

2.6.2 State Estimation Results

As far as the state estimation algorithm is concerned, our target is to prove that our 3SMA-RPi is able to perform each step of the algorithm as efficient as the 3SMA-Laptop unit. We also want to prove that our 3SMA-RPi is able to complete each step in less than 1 minute, as this is the time occurring between two consecutive states evaluations. As in the previous section, to ensure a more realistic condition, we included in our setting the data network simulation to test the algorithm with different levels of network congestion.

Our first test focuses on verifying that the infrastructure and our two 3SMA prototypes are able to communicate, run the distributed algorithm, and update the MQTT topic in case of a network reconfiguration. The state estimation has been verified using both our prototypes running in two possible configurations, i.e., with 3SMA-RPi executing the LV grid-level SE and the 3SMA-Laptop running the MV grid-level SE, and vice-versa. This enables a fair and head-to-head comparison in terms of performance and accuracy.

As shown in Figure 2.13, when 3SMA-RPi (M2 in Figure 2.7) runs the LV grid-level SE, it takes about 11.9±0.4 s for each computation. When it executes the MV grid-level SE (M1 in Figure 2.7), it takes 4.13±0.03 s. Repeating the very same tests with the 3SMA-Laptop unit, the two steps are executed in 0.127±0.003 s and 0.057±0.003 s, respectively. As the same version of the algorithm is used on both units, the accuracy achieved is the same and we did not notice any difference in terms of outputs accuracy. Thus, we can claim that our 3SMA-RPi unit is perfectly suitable to address such computation, even if it takes a longer time for its execution. To better analyze our infrastructure, as in the previous cases, we need to consider the latency time added to the process by the communication network. Indeed, as introduced in the previous sections, the MQTT is used to send partial results from the smart meter performing the LV grid-level SE to the one running the MV grid-level SE algorithm. Since the size of the messages is quite similar to the one of FLISR (i.e., 1.4 KByte), the maximum possible latency in the worst case is less than 1 s, as shown in Figure 2.10. Since the proposed procedure runs a state estimation algorithm every minute, the 3SMA-RPi unit is fully compliant with this requirement as it is able to complete all phases, i.e., LV SE and MV grid-level SE, in less time. Moreover, in case of a fault, after the topology changes occurred during the restoration phase, the MQTT topic and the configuration of the SE algorithm are correctly updated according to the portion of the MV network monitored by the 3SMAs. As discussed in Section 2.6.1, this reconfiguration can last about 10.5 s. During the network reconfiguration, in case one of the 3SMAs deployed across the grid is disconnected from the Internet, it will receive the new setting when the next auto-update routine starts. The time period between two consecutive auto-update routines depends on the end-users settings configured in the DeC (see Section 2.4.1).

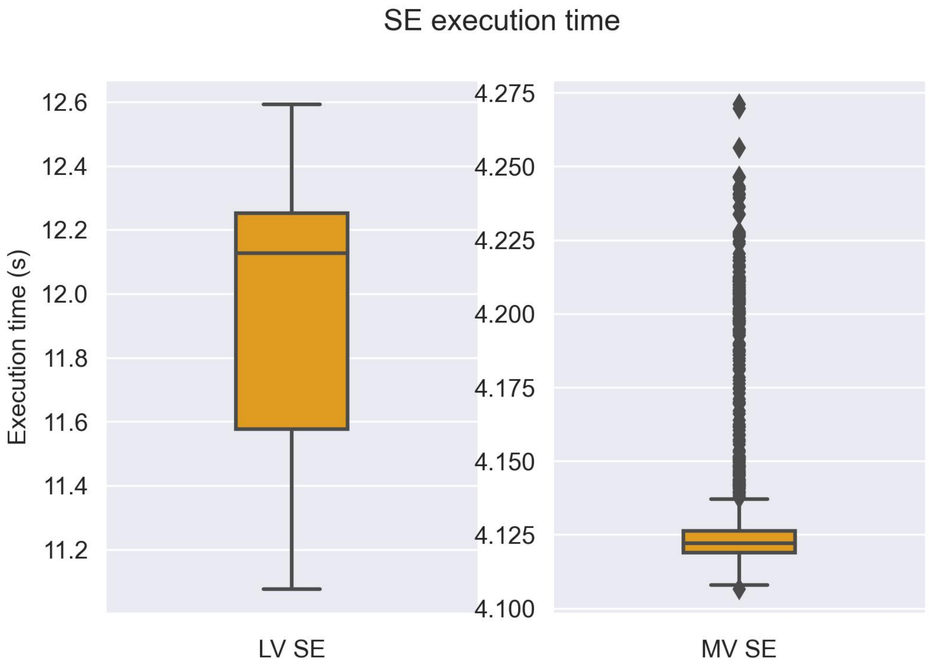


Fig. 2.13 Execution time of the state estimation for both LV and MV grid-level SE running on our 3SMA-RPi unit.

When our algorithm runs on the 3SMA-RPi, even if the memory usage is just at 3.9%, the CPU usage is always around 100%, with some outliers that even exceed that value. This result may look liks an issue, but there are two main factors to take into consideration:

- The algorithm is written in Matlab© and it runs on 3SMA-RPi using Octave, thus, the code is not optimized at all for embedded devices.
- The target of our test is to assess the possibility to run a distributed algorithm on the 3SMA prototype, not to optimize our application for real usage.

Once these two considerations are correctly taken into account, our results look promising. Even without any code optimization and with the adoption of low-cost hardware devices, the 3SMA is perfectly able to run the different levels of SE and to satisfy the time threshold required by the algorithm. In conclusion, using inexpensive hardware devices reduces costs and improves scalability but it also implies the necessity to carefully design and optimize the applications deployed on the devices.

2.6.3 Final Remarks on Communication Latency

Assessing the impact of data transmission in existing communication networks is crucial to address novel service requirements in terms of data transmission latency as well. Indeed, such latency can affect the operational status of the smart grid. The IEC 61850 standard [7] defines the communication requirements to be addressed in power distribution networks. Table 2.7 reports the performance classes defined by the standard. Our experimental results on time delay, performed by exploiting Mininet to simulate a realistic virtual MAN, satisfy the requirements of classes *TT0*, *TT1*, and *TT2*. This is confirmed when the MAN congestion rate is lower than 90%, as in this case, the maximum latency never exceeds 500 ms. When the congestion rate is 100%, the median and the maximum latency values are about 350 ms and 1 s, respectively. However, as previously discussed, this latter scenario is very critical and must be avoided because it could lead to the collapse of the MAN itself.

2.7 Conclusion and future work

In this chapter, we propose a low-cost smart meter architecture and a distributed software infrastructure for AMI, which can collect data from the network, communicate with other entities, and offer different features to each actor connected to the network. The usage of well-known IoT technologies ensures a high compatibility with other devices and third-party services. Our 3-phase meter architecture is able to run multiple software applications, either on-board or distributed over the network, and to auto-update its status when required. Moreover, new algorithms can be added, updated, or removed on the fly thanks to its auto-configuration capability. This ensures high compatibility with other device management tools and communication

Performance Requirements	Performance Classes	Values [ms]	Example of services
Transfer time	TT0	>1,000	Files, events log contents, SCADA
	TT1	1,000	Events, alarms
	TT2	500	Operator commands
	TT3	100	Slow automation interactions
	TT4	20	Fast automation interactions
	TT5	10	Releases, status changes
	TT6	3	Trips, blockings

Table 2.7 The IEC 61850 [7] standard: Communication requirements and performance classes for power systems.

protocols such as DLMS/COSEM [33], SML [34], and IEC 61850 [7], which could be added and treated as any other data processing algorithm.

To verify the characteristics of the entire infrastructure, we run real-time simulations with different configurations and settings. The experimental results prove that the system can identify and locate grid problems at high speed, restoring the original functionalities faster than any other state-of-the-art solution. Moreover, the time delay on data transmission, estimated by including a MAN simulator in our realistic test-bed environment highlights that the resulting latency respects the limits imposed by the IEC 61850 standard [7].

Among the possible future work, we would like to mention that the proposed distributed software platform and the 3SMA will be integrated in a wider distributed multi-model co-simulation environment [73]. The target of this effort is to unlock other possible scenarios and test additional multi-energy services such as optimal management of renewable energy sources. In addition, we also plan to further optimize our algorithms and software running on our 3SMA to improve their performances when implemented on embedded systems with reduced computation power.

Chapter 3

Photovoltaic panel:technology background and investor opportunities

This chapter is devoted to giving a brief explanation of the PV technology and the new actors that will participate in the transition toward the Smart Grid. Section 3.1 explains how the PV modules technology works and which are its limitations, Section 3.2 explains how the prosumers and the REC can use such technology in the smart grid and explains the role of the EA. Finally Section 3.3 and Section 3.4 introduce the common methodologies that have been used for the works presented in the following chapters.

3.1 Photovoltaic power generation

3.1.1 PV hierarchy

The basic element of a PV generator is the *cell*, whose behaviour can be described by an ideal current source, proportional to solar irradiance, and by a diode connected in anti-parallel. A cell is described by a voltage-current (I-V) characteristic curve (right of Figure 3.1, black lines), which, at a given cell temperature, changes as a function of the irradiance G: when G increases, the open-circuit voltage V_{oc} increases logarithmically and the short-circuit current I_{sc} increases proportionally.

With fixed irradiance G, a temperature increase yields a slight increase of I_{sc} and decrease of V_{oc}. The maximum of the corresponding voltage-power (P-V) curves (grey lines) corresponds to the optimal conditions for extracting power, given the current irradiance.

PV installations are organized hierarchically (left of Figure 3.1): cells are connected together into a PV *module*; PV modules can be further interconnected to form a *PV array* to achieve the desired voltage and current levels. The typical connection is organized as a number of parallel strings, each composed of the same number of PV modules connected in series.

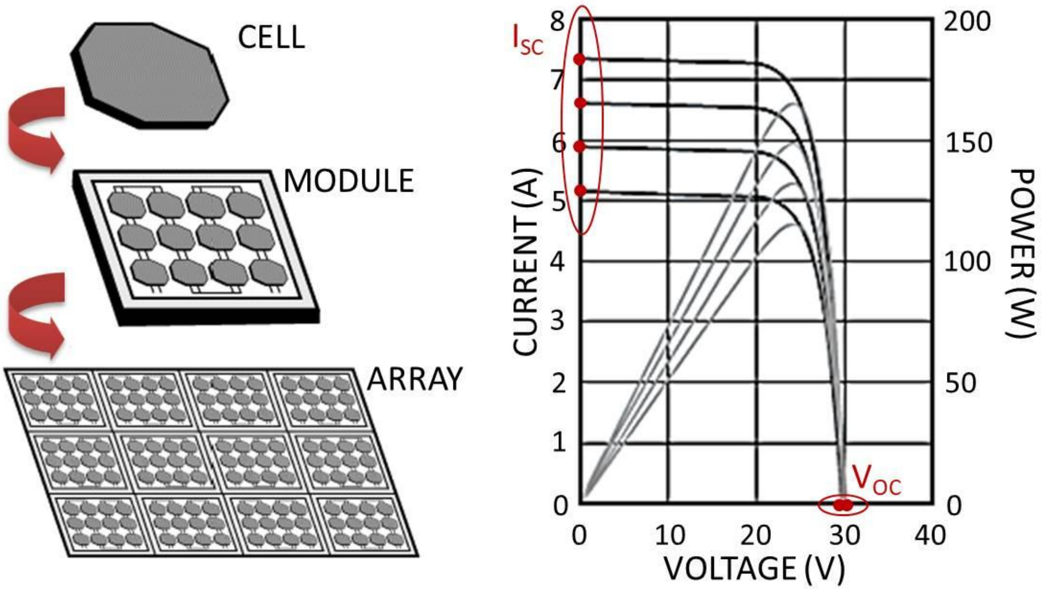


Fig. 3.1 PV hierarchy (left) and voltage-current (I-V) and voltage-power (P-V) characteristic curves of the Mitsubishi's PV-MF165EB3 PV module [2].

3.1.2 Impact of uneven irradiance distribution

As anticipated in Section 3.1.1, irradiance has a heavy impact on the power production of a PV module. PV installations are typically designed by assuming that the surface of interest is subject to even irradiance. This is however not accurate, as shadows projected by obstacles such as chimneys, surrounding buildings, trees, etc determine a heterogeneous distribution of irradiance, with the effect that each PV module will operate at different irradiance conditions. Shading is critical for PV installations, as the least irradiated PV module acts as a bottleneck on the power production of the overall installation: the higher the variance of irradiance, the higher the power loss. When PV modules are connected in series, the least irradiated module will provide the smallest current, thus restricting the available current of the string; when series strings are then connected in parallel, the string with lowest voltage will determine the voltage of all strings. This leads to a potentially high power dissipation, resulting in local overheating, accelerated ageing and permanent cell damage [74].

3.2 Economic aspect of RES owners

In the Smart Grid, the customer will change by transforming in the so-called *prosumers*, i.e. those users that are able both to withdraw or inject energy in the grid. As discussed in Section 1.3 the most common technology that will be chosen by the prosumer will be the PV due to its relatively small cost and the possibility to install it without major difficulties even in urban environments. In addition to that multiple users will create Renewable Energy Communities. Those communities can be seen as a group of customers that organize to become all prosumers by sharing the cost and the benefit of a common Renewable Energy System (can be PV or others).

3.2.1 Prosumers and REC finance

However, given the economic effort needed to instantiate and maintain a new PV system, it is extremely important to analyze the various financing schemes available to choose the best one according to each scenario.

- Self-funding

  This is the simplest and most common scheme. In this scenario, the system owner, prosumer or REC, uses his own money to pay for the system beforehand. This schema during the last years has been the most common end effective for small-scale residential and commercial customers. The main limit of this approach is that is suitable only for those users that have enough money already at their disposal.

- Debt

This is the other most common schema for the funding of renewable energy systems. As the name suggests with this approach the owner of the system borrows a portion of the entirety of the money necessary to acquire the system. However, the borrowing of money can assume different forms: - Personal loan: in this case banks or other credit agencies will lend money to the owner to pay for the initial cost of the system - Project finance: in this scenario, the earnings generated by the system are used to pay back the initial loan. In addition to these, other debt-based solutions have been adopted by each country such as the transfer of tax credits etc.

- Leasing

  This financing scheme is very useful to create a larger RES plant with respect to the one that could be deployed by single prosumers or small REC. In this approach, a company takes care of the planning and purchasing of the PV modules and installing the system on the customer's roof. In order to use the energy provided by the roof it is required to pay a rent fee to the company. This approach allows the REC participants to avoid the high initial investment needed to deploy the system, however, unless differently stipulated in the contract with the system owner, they still need to take care of the maintenance cost of the PV panels.

3.2.2 Operational approaches

Once the PV system of a REC or a prosumer is installed, different approaches can be used to define how to use the energy produced by the system and how to manage the possible surplus. Therefore different business models can be adopted, the choice however is often influenced by the law regulation and by the grid flexibility. The most common business model assumes that the participant to the REC (or even the single prosumer) self-consumes most of the energy produced by the system avoiding the need to buy it from the grid. For the surplus of energy, there are two main possibilities: store the excess for further use when the system reduces its production (for example during nighttime in the case of a PV plant) or reintroduce the energy into the main grid in exchange for some kind of benefits. In some cases, when it's allowed by regulation and technologies these two approaches can also coexist and the REC can decide whether it is more convenient to store, sell or buy the energy.

Regarding the benefit that the REC could receive from the introduction of energy in the main grid, the strategy usually applied is called *net metering*, which involves two possibilities:

- *reverse metering*: the system operator will reduce the amount of energy that the prosumer/REC needs to pay,
- *financial credits*: the prosumer or the REC will receive credit for buying energy in the future

The selection of one among those two approaches mainly depends on the legislation of the country in which the prosumer or the REC is located and or whether the grid is capable or managing the energy introduction.

3.2.3 EA or Virtual Power Plant

While the prosumers and the REC approach are two non-profit approaches, there is another approach that will be diffused in the Smart Grid. It can be seen as the direct counterpart of the leasing approach and it requires the introduction of a new actor in the energy market, the *EA*. In this case, it will pay a rental fee to install the PV system on the roofs of the customers. As shown in Figure 3.2, its role is to be in the middle of customers and distribution system operators by aggregating the demand of

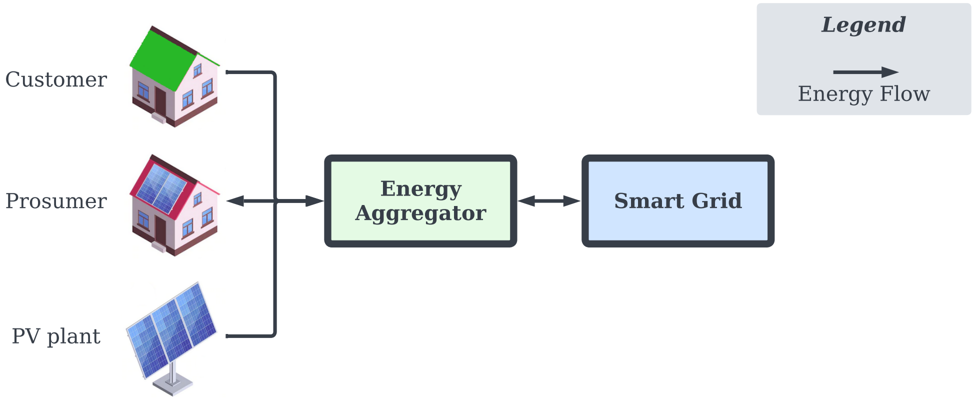


Fig. 3.2 The EA is an actor that aggregates the energy demand and production of the customers and manages the interaction between users and the grid

multiple customers. This operation allows the aggregator to buy energy at a reduced cost from the DSOs or even earn by selling them the surplus produced by the panels. So by playing with the prices at which it buys or sells energy to the DSOs and the price at which it sells the electricity to the customers it is able to generate profits.

3.3 GIS methodology for PV placement

In all of the described operational approaches, the placement of PV systems in buildings' roofs should be carefully designed to maximise both the power production and, consequently, the ROI. To this extent, it is necessary to avoid any source of inefficiency: PV modules must be installed in areas with the best conditions in terms of solar irradiance, so to limit the impact of shading and the consequent power losses. At the same time, it is crucial to take into account the economic dimension, to guarantee that the initial investment is compensated by revenues over time, and and adequate ROI. So far, this has been performed by following some general methodologies that try to roughly estimate these two parameters based on rule-of-thumb criteria. To achieve better results the stakeholders need the fine grain data of the irradiance of the zone or the rooftops they want to utilize in order to obtain a better placement. In some cases, this information is used to obtain which are the surface that can be used to place the PV panels but these does not provide any information on how to actually place it. There is also the need to take into account the economic aspect: some stakeholders may be interested in placing a given amount of panels while others may have budget limitations.

My Ph.D. activity proposes a novel GIS-based methodology for an optimal configuration and placement of PV systems in building roofs. It exploits detailed information on air temperature and solar irradiance to determine the optimal series-parallel topology of a given number of PV modules in a target area on the roof, evaluating also the foreseen payback time of the installation. This methodology has been included in a more complex framework that allows to analyze of three different scenarios: a single prosumer, an Energy Aggregator and a Renewable Energy community. Despite the common goal of having an optimal pv placement, each operational approach needs to be addressed on its own due to the opportunity and the limitation that characterize them. For example, the way in which the panels can be connected and the methodology used to evaluate the energy production have major differences that have an impact on the economic analysis.

The common denominator of those three scenarios is the methodology used to identify the suitable areas and the identification of the most irradiated portion of those areas that are the most promising for the installation of pv panels. The following subsections are devoted to the explanation of such methodology that will be used in the following chapters.

3.3.1 Suitable area identification

Starting from a DSM of the area of interest, the algorithm identifies possible encumbrances and the corresponding evolution of shadows to find areas that could be used for the deployment of the PV installation.

To process theDSM , we used GDAL[75], a translator library for raster and vector geospatial data formats. This allowed to identify the surfaces (i.e., roofs) that maximize power production in terms of tilt angle and orientation, depending on the geographic location of the district. In our case study, we generated two raster images in GeoTIFF format: one to store the data describing the roof slope and the other to store the data of the aspect for the area under test. Then we processed these two files to extract the surfaces with a slope between 15 and 36 degrees and aspect value between 240 and 300 degrees in order to have roof pitches oriented towards South, configuration that guarantees optimal sun exposition and potential PV production [76]. Top of Figure 3.3 shows an example roof (pink) to highlight how the area that it is suitable for PV installation resulting from this step (purple) is only a portion with respect the whole area.

Each identified suitable area is then annotated with the inclination and the aspect of its roof pitch. Those areas are then intersected with cadastral maps to annotate the average height of the roofs. This information is extremely useful as the height difference of contiguous roofs must be taken into account when the placement algorithm is executed, to allow installation of connected PV modules on different roofs without incurring in high dispersion of the produced power.

The areas are then sampled with the same resolution of the DSM (i.e. $1m$). The result of this sampling is shown in top of Figure 3.3, where each point (yellow) corresponds to a pixel of the DSM model that resides within the area. Using these points,

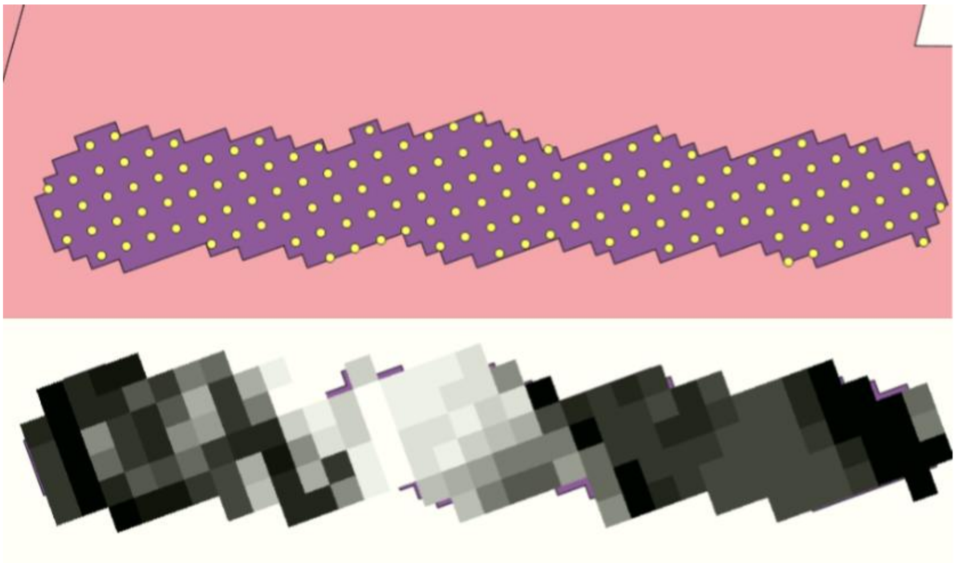


Fig. 3.3 Example of total area of a roof (pink), area suitable for PV module placement (purple), and DSM points covered by the area (yellow) (top), and corresponding 75th percentile over one year of the suitable area (bottom, darker areas are subject to more shading and thus to lower irradiance).

we are able to determine the evolution of irradiance over time from yearly weather data by using the shadow model developed in [77]. Using georeferenced points allows to generate and store data only for the areas that could actually be exploited for the installation of the PV modules. This is a key point of the proposed approach, as it allows to optimize the amount of data that must be generated, memorized and handled thus guaranteeing an optimal trade-off between the size of the district of interest (that can be of a number of squared kilometers) and achieving at the same time a fine-grain representation of the solar evolution of the suitable areas over time.

3.3.2 Performance pre-evaluation

The next step consists of identifying the most irradiated portions of the suitable areas, that are the most promising positions for the installation of PV modules. The suitable area is explored by considering all possible placements of a PV module inside of the area. The yearly trace of irradiance of a position is derived from the DSM by considering the traces for the DSM point covered by the PV module. In case there are multiple points, for each time instant we consider the minimum value of the irradiance over all the traces, to reproduce the bottleneck effect described in Section 3.1.

To get a compact signature and allow easy comparison of positions, we use the *75th percentile of irradiance*, i.e., the value below which 75% of samples of a yearly irradiance traces fall. The 75th percentile thus allows to discriminate between highly irradiated and poorly irradiated positions, as larger values of the percentile identify positions whose irradiance trace distribution is more skewed towards the upper range of the irradiance values, i.e., that are more irradiated and more promising for PV power production. Figure 3.3 exemplifies this process by showing a suitable area and its heat-map, where whiter points corresponds to a higher value of the 75th percentile (i.e., more irradiated points).

The PV module positions are then sorted by decreasing value of the 75th percentile. The user is now allowed to give a threshold value $minTh$ representing the minimum value of the percentile that is accepted for PV placement. The experimental analysis will prove the impact of this choice on the resulting identified solution. All positions with 75th percentile lower than $minTh$ are now excluded from the suitable area, as the performance indicator identifies them as non-promising locations from the perspective of PV power generation. The resulting suitable area represents the area that can be occupied by PV modules to achieve optimal power generation.

3.4 Power production estimation

The estimation of the power of an individual single PV module, because the total power extracted by the panel P_{panel} depends on its actual series/parallel topology and is in general different from the sum of the power of the individual modules. Given a *mn* series-parallel interconnection (i.e., n parallel strings each of m modules in series), the total power is obtained as $P_{panel} = V_{panel}I_{panel}$, where:

$$\begin{cases} V_{panel} & = & min_{j=1 \to n}(\sum_{i=1 \to m} V_{module,ij}) \\ I_{panel} & = & \sum_{j=1 \to n}(min_{i=1 \to m} I_{module,ij}) \end{cases} \tag{3.1}$$

and $V_{module,ij}$ and $I_{module,ij}$ are the voltage and current extracted from the i-th module in the j-th series.

The following Chapter 4, 5 and 6 will explain three different solutions designed to assess the different needs of each of the actors introduced in this chapter by using the PV technologies and taking into account both the opportunities and the constraints that each possibility involves.

Chapter 4

Optimal configuration of the placement of PV panels from a prosumer perspective

This work proposes a framework to achieve an optimal PV installation, both in terms of power production and ROI. The key enabling technology is a GIS-based approach to generate irradiance and temperature evolution over the roof over one year. Such fine-grained and detailed information allows determining which areas of the roof are more promising, as less affected by shading caused by encumbrances (e.g., nearby buildings, trees, dormers). To minimize power losses caused by partial shading and by an inefficient connection of the PV modules, we additionally analyse the correlation between the irradiance evolution over time of different areas of the roof, to guarantee that bottleneck effects are if not completely avoided at least limited.

This leads to the definition of an effective exploration over a number of possible configurations: for each configuration, we determine the optimal placement and connection of PV modules, trying to maximize irradiance and to minimize sources of inefficiencies. The irradiance and temperature data are then used to estimate the yearly power production of each installation, and to derive the corresponding ROI and the PT of the investment. The result is the identification of the optimal configuration, in terms of both power production and revenues.

To prove the effectiveness of the proposed framework, we applied the analysis to two roofs of industrial buildings. Results show an improvement on power generation from 11% up to 23% w.r.t. standard installations. Moreover the outcome highlights that a cost analysis, typically ignored by standard installation strategies, is crucial to determine the optimal installation, as more PV modules do not always guarantee the highest ROI.

The rest of the chapter is organized as follows. Section 4.1 reviews relevant literature solutions on this topic. Section 4.2 presents the proposed methodology. Section 4.3 discusses the experimental results. Finally, Section 4.4 provides the concluding remarks.

4.1 Literature review

GIS are recognised as a key technology in modelling the evolution of solar irradiance and in simulating PV production [78]. GIS can also be used to plan installation of PV systems in urban environments [79]. For these purposes, such GIS-based solutions take advantages of *DSM* or 3D city models given by LiDAR data. DSM is a geo-referenced raster image representing the terrain's surface with all objects on it and elevation.

Efficient PV design and installation has been deeply investigated in the literature. In this scenario, solutions like [80–85] exploit both geographic data and evolution of shadows to estimate PV power production. However, they abstract the modelling of PV power production both in terms of model accuracy and placement topology of PV modules. In addition, these literature solutions do not integrate sub-hourly meteorological data needed to better estimate PV power production in real-sky conditions. PVGIS [82] and PVWatts [81] are the only exceptions performing hourly and sub-hourly simulations, respectively. However, they use a low-resolution DSM (i.e. $> 1m$) that is not suitable to identify encumbrances in roofs like chimneys and dormers. To achieve that, a DSM resolution smaller than $1m$ is needed. Moreover to perform realistic and accurate simulations, real meteorological data collected by weather stations [79] have to be used to compute the incident solar radiation on roofs in real-sky conditions. Finally, none of presented solutions provide guidelines for a smart, GIS-driven floorplanning of PV modules.

Other approaches focus on the optimal sizing of the PV installation [86–89], on the identification of optimal management algorithms [90, 91], or on optimal roof configuration (e.g., tilt angle) [92], by abstracting on the other aspects of PV designs. Vice versa, the works in [93–95, 80] focus on the identification of suitable areas at large (i.e., entire roofs or geographical areas), with no detailed information about the actual placement of PV modules and with very abstract models of PV power production. The work in [3] proposes an algorithm to exploit environmental traces to determine an optimal placement of PV modules. However, the traces are used only to determine heatmaps of the most promising locations on the roof, and it considers the configuration of the PV installation as a user-defined input, with no exploration of potential alternatives based on a cost analysis.

With respect to the literature, the approach proposed in this chapter explores a number of alternative configurations and identifies the optimal one both in terms of power production (through a placement of PV modules that is aware of fine grain environmental traces) and of economic investment (highest ROI and lowest PT). Additionally, this work extends [3] with an analysis of the correlation of irradiance over time of different areas of the roof, to further minimize bottleneck effects.

4.2 Methodology for placement optimization on single roofs

The goal of the chapter is to *determine an optimal configuration for a PV array through a design space exploration of a number of possible alternatives*. We first generate traces of irradiance G and temperature T over time for the roof of interest, with a fine grain time and spatial granularity (A). Then, we determine a number of configurations of interest from user-defined input (B). For each configuration under analysis, we estimate an optimal placement of PV modules on the roof (C), estimate the corresponding yearly power production (D) and evaluate the foreseen payback time of the installation (E). This will allow to easily identify an optimal configuration for the roof of interest (F), aware both of environmental and cost variables.

4.2.1 Irradiance and temperature traces for the roof of interest

The proposed PV floorplanning algorithm takes advantage of fine-grain resolution maps of irradiance and temperature, generated by exploiting the software infrastructure built in [77]. Given a high-resolution DSM of the roof of interest, the software identifies encumbrances (e.g. chimneys and dormers) and estimates the evolution of possible shadows. With such information, it is possible to perform the identification of the suitable area of the roof that can be used for the placement of PV panels, that is then aligned to a *virtual grid* with fine grain space granulatity (20cm), used by the PV placement algorithm. The evolution of irradiance over time is obtained by combining weather data, retrieved from personal or third-party weather stations [96], with the shadow model. Weather temperature and incident irradiance are then used to determine the distribution of temperature on roof over time, as an effect of convectivity and radiative loss [77]. The result are a set of measures of irradiance $G(i, j,t)$ and temperature $T(i, j,t)$, each associated with a grid cell in position (i, j) at a given time step t, with 15 minutes granularity.

4.2.2 Available configurations

Each configuration considered in the exploration is identified by the number of PV modules N_i to be placed on the roof. All configurations have the same connection topology: they are organized as strings of s PV modules in series, with s given in input by the user. Thus, different configurations vary in terms of number of strings connected in parallel $p_i = \frac{N_i}{s}$. The number of modules is limited by a maximum N_{MAX}:

- *user-defined*: the user gives in input the maximum number N_{MAX} of PV modules that can be explored;
- *cost-defined*: the user specifies a maximum budget B that can be invested and the cost *unit_cost* of a single PV module of interest; thus $N_{MAX} = \frac{B}{unit\_cost}$.

If N_{MAX} is too large for the suitable area, N_{MAX} is set to the maximum number of PV modules that can be hosted.

4.2.3 Optimal PV placement

Given an input configuration $N_i = s \times p_i$ of PV modules to be installed, the goal of the placement algorithm is to exploit the environmental traces $G(i, j,t)$, $T(i, j,t)$ to identify an optimal placement and connection of the PV modules on the roof. Given that an exhaustive exploration of the possible solutions is not feasible for roofs of a reasonable size [3], the algorithm builds an approximation of the optimal solution based on a suitability metric.

Suitability metric The most promising positions for a PV module are those guaranteeing a higher irradiance over time. The suitability metric $S(i,j)$ of a cell position (i,j) should thus give an aggregate measure of the potential PV production that can be achieved when placing the top left corner of a PV module in (i,j). The suitability metric $S(i,j)$ is the *75-th percentile* of irradiance, that represents the value below which 75% of the samples of $G(i,j,t)$ fall for grid position (i,j): larger values of the percentile identify distributions that are more skewed towards the upper range of the values. The suitability metric $S(i,j)$ is then applied a corrective factor, modeling the impact of temperature[1].

Similarity metric Connecting PV modules with similar irradiance over time is especially important when connecting PV modules in series[2]: e.g., the placement algorithm must avoid the series connection of PV modules that have similar suitability metric but that are on the opposite sides of an obstacle, with the result that when the one is in full sun the other is shaded, and vice versa. A measure of similarity between two grid cells (i,j), (i',j') is given by the correlation of their irradiance traces over time, i.e., a measure of how linearly dependent the traces of $G(i,j,t)$ and $G(i',j',t)$ are over time. This is estimated by calculating the Pearson correlation coefficient of the irradiance traces: a high correlation coefficient ensures that two PV modules placed in (i,j) and (i',j') would behave similarly.

PV placement algorithm The resulting PV placement algorithm is given in Figure 4.1. First (line 1) the suitability matrix S is computed as described above for all grid cells. Grid cells are then sorted in decreasing order of S in vector V: the first cells are the most promising ones (i.e., higher 75^{th} percentile of irradiance). We then iterate the choice of the placement of the Ni PV modules to be placed in

[1] Temperature is considered only as a corrective factor as its impact on power production is limited w.r.t. irradiance (less than 0.5%/°C) [2].

[2] The bottleneck effect is heavier in case of series connection, as module voltage has a weaker dependence on irradiance than current [97].

series-first order, i.e., modules belonging to a series string are enumerated before moving to another string. When choosing the position of the first PV module of a series string (lines 5-8), we pick the grid cell with highest suitability metric (i.e., $V[1]$). When choosing the placement of the subsequent $s-1$ PV modules of the string (lines 10-18), we must take into account the correlation of their irradiance traces w.r.t. the first PV module of the series ($first$). To this extent, we introduce a threshold th: given the grid cell with highest suitability matrix $V[l]$, it is added to the series only if its correlation w.r.t. $first$ is higher than th. Else, the algorithm moves to evaluating the next position. Note that any time the placement of a PV module is chosen, all grid cells covered by the module become unusable and are thus removed from vector V. The loop terminates when the Ni PV modules have been placed.

4.2.4 Estimation of yearly power production

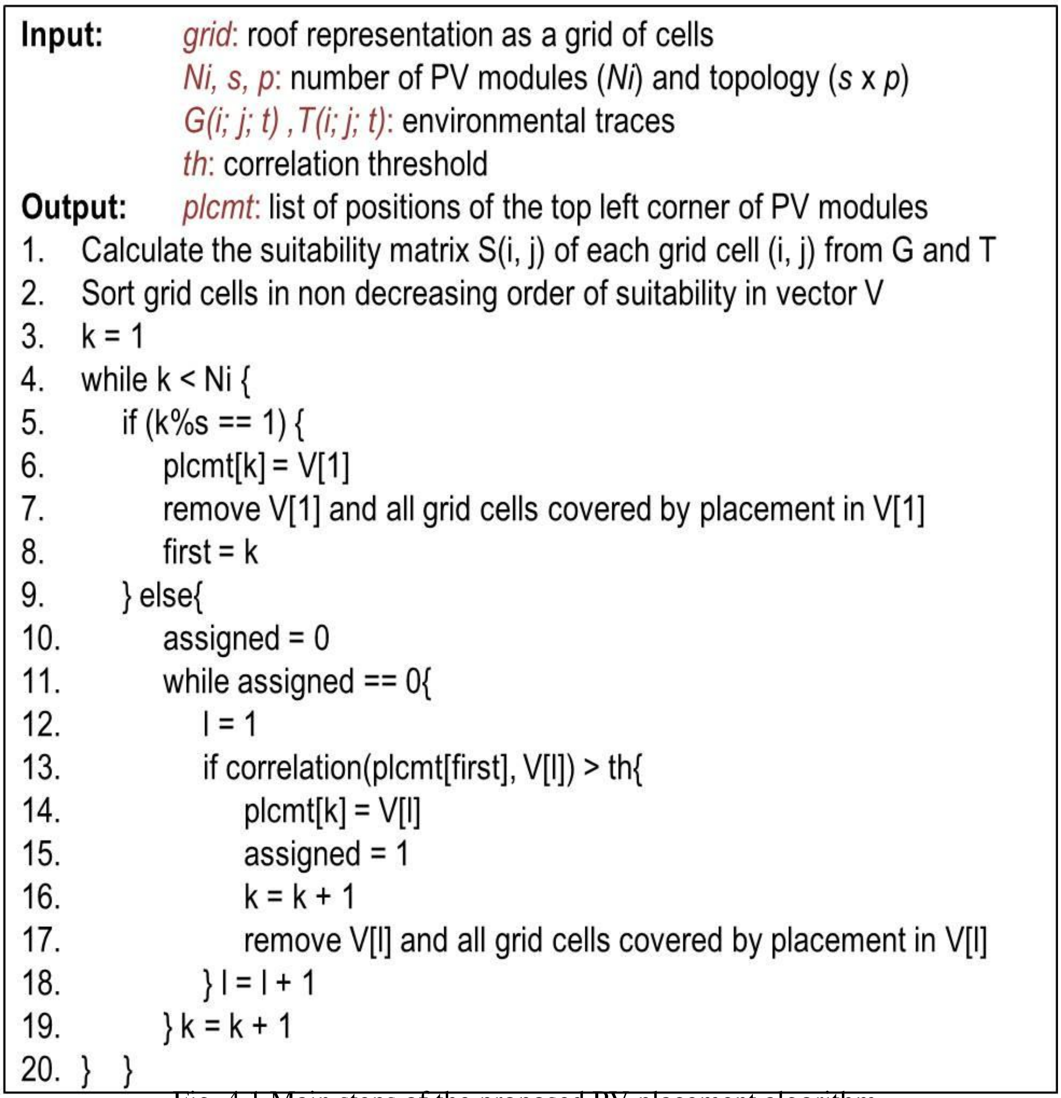
Given the placement of PV modules, it is necessary to derive the corresponding yearly power production.

PV module model

The power production of each PV module is determined with the model proposed in [3, 98], that exploits information available in public datasheet of the PV module to build a simple linear model. Power dependency on G and T is determined from the thermal and irradiance coefficients and the I-V graphs in the datasheets, respectively. The result are linear equations of current $I(i,j,t,G,T)$ and voltage $V(i,j,t,G,T)$, that linearly depend on the value of the environmental traces $G(i,j,t)$ and $T(i,j,t)$ in the given PV module position (i,j) at current time t. The linear models ensure fast construction and simulation time, thus reaching a good accuracy/speed trade off, crucial in a design space exploration, and allowing the evaluation of the yearly PV production of a high number of alternative configurations.

Topology application

Given the individual production of PV modules, the impact of their $s \times p$ series and parallel connection is as follows. We first consider the *series connection* of PV modules composing a string, where the PV module with the lowest current (i.e.,



```
Input:    grid: roof representation as a grid of cells
          Ni, s, p: number of PV modules (Ni) and topology (s x p)
          G(i; j; t) ,T(i; j; t): environmental traces
          th: correlation threshold
Output:   plcmt: list of positions of the top left corner of PV modules
Calculate the suitability matrix S(i, j) of each grid cell (i, j) from G and T
Sort grid cells in non decreasing order of suitability in vector V
k = 1
while k < Ni {
    if (k%s == 1) {
        plcmt[k] = V[1]
        remove V[1] and all grid cells covered by placement in V[1]
        first = k
    } else{
        assigned = 0
        while assigned == 0{
            l = 1
            if correlation(plcmt[first], V[l]) > th{
                plcmt[k] = V[l]
                assigned = 1
                k = k + 1
                remove V[l] and all grid cells covered by placement in V[l]
            } l = l + 1
        } k = k + 1
}   }
```

Fig. 4.1 Main steps of the proposed PV placement algorithm.

irradiance) limits the current of the other PV modules, while string voltage is the sum of single PV module voltages. Then, we take into account the *parallel connection* of strings: the PV string with the lowest voltage constraints the voltage also of the other strings, while the current is the sum of currents generated by the single strings.

Yearly power production

Yearly power production P_{yearly} of the PV installation is estimated by applying the aforementioned formula to the traces of G and T over time. By accumulating power over all time samples gives thus an estimation of the yearly power production of the installation:

$$P_{yearly} = \sum_{t=1...T} V_{array}(t) \cdot I_{array}(t)$$

where T is the length of the environmental traces (i.e., 35,040, the number of 15-minutes samples in a year).

4.2.5 Cost analysis

In this work we use two metrics of the effectiveness of a configuration. The ROI is a measure of efficiency of an investment:

$$ROI = \frac{yearly\ revenue - cost\ of\ installation}{cost\ of\ installation}$$

and it allows to compare the different configurations according to the gains provided in one year of operation. PT is an indication of the amount of time that is needed to payback the initial investment:

$$PT = \frac{cost\ of\ installation}{yearly\ revenue}$$

In both formulas, the cost of installation is given by the unit cost of a PV module multiplied by the number of PV modules in the current configuration. The yearly revenue takes into account the amount of money earned by selling the generated power to the grid (calculated by multiplying the yearly kWh production of a configuration P_{yearly} per the price of energy E_p), minus the cost of maintenance M_c, originated by the periodic cleaning, monitoring and repairing of PV modules (e.g., panel damage, fractures, and frame corrosion) [99]. E_p and M_c are provided in input by the user.

4.2.6 Identification of optimal configuration

Given the yearly power production, the ROI and the PT of each configuration identified in Section 4.2.2, the optimal configuration is selected as the one that maximizes the ROI, while minimizing the PT. Notice that the solution with the highest number of PV modules may not be the optimal one, due to an increase not only of power production but also of the initial investment and of maintenance costs.

4.3 Experimental results

4.3.1 Simulation setup

We applied the proposed algorithm on two lean-to roofs, reported in Figure 4.2 (black areas represent to the presence of encumbrances that do not allow PV module installation, e.g., pipes). Using the strategy in Section 4.2.1, we first generated the values of irradiance and temperature over one year. Figure 4.2 shows the 75^{th} percentile of irradiance over the roof, where the clearer cells are the more irradiated (i.e., highest suitability metric). The heterogeneity in color distribution highlights the high variance of irradiance distribution, despite of the relatively small size of roofs (roof 1 is 49m×12m, roof 2 is 42.8m×12m). In our setup, we consider a PV-MF165EB3 module by Mitsubishi [2], and the configurations of interest are generated by considering strings of $s = 8$ PV modules and $N_{MAX} = 72$, defined as user configuration.

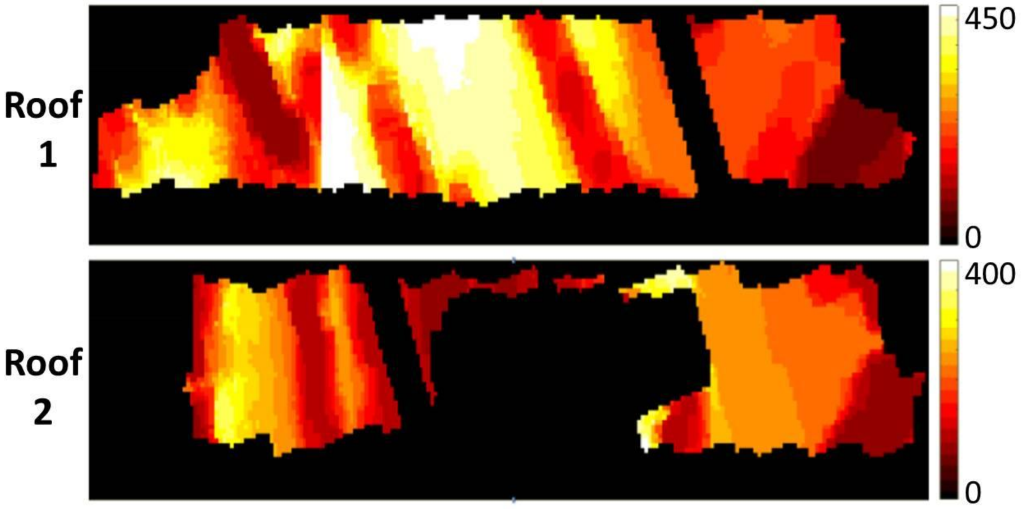


Fig. 4.2 75^{th}-percentile of G over the roofs (the clearer the more irradiated).

4.3.2 Analysis of the proposed PV placement

To better evaluate the performance of the placement algorithm proposed in Section 4.2.3, we focus the analysis on two configurations per roof, i.e., with $N_i = 32$ and $N_i = 72$. Figure 4.3 compare three placements per each roof: a traditional

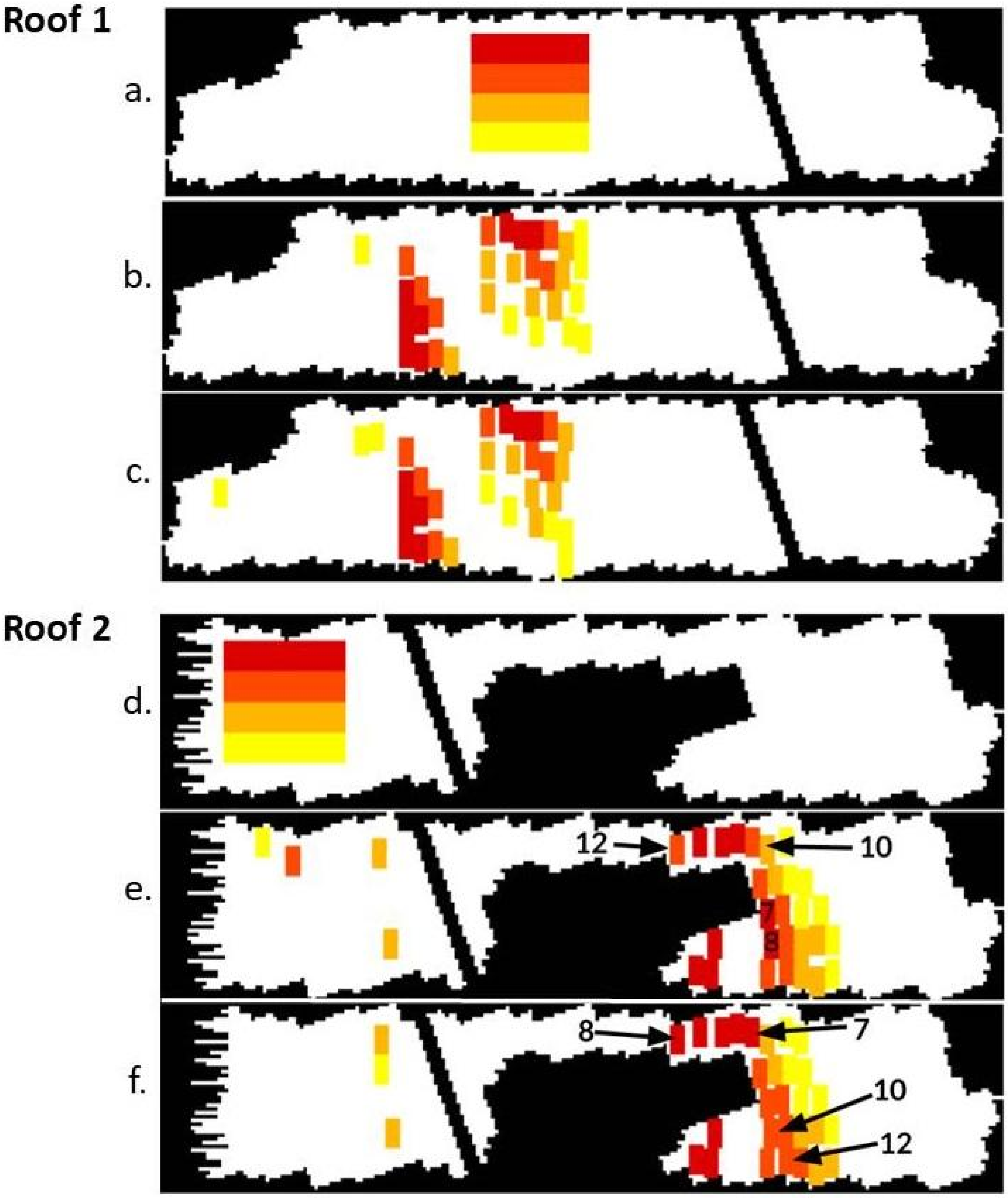


Fig. 4.3 Possible configuration for N=32 for roof 1 ($a-c$) and roof 2 ($d-f$): traditional compact placement (a,d), placement generated by [3] (b,e), proposed algorithm (c,f). Rectangles are PV modules, rectangles of the same color are PV modules connected in series.

compact placement (a,d), the placement generated by [3] (b,e), that only observes the distribution of irradiance on the roof, and the placement generated by the proposed algorithm (c,f). Table 4.1 reports the yearly power generation of each configuration, plus the improvement of [3] and of the algorithm proposed in this work (with threshold set to 0.9, i.e., correlation 90%) w.r.t. the traditional placement.

Roof	N	Traditional	[3]		Proposed	
		(kW)	(kW)	(%)	(kW)	(%)
Roof 1	32	6.21	6.94	11.70	7.01	12.80
	72	12.19	14.27	17.10	14.66	20.20
Roof 2	32	5.04	5.28	4.80	5.60	11.10
	72	9.37	11.25	20.00	11.57	23.00

Table 4.1 Yearly power production of [3] and of the proposed algorithm with respect to the traditional placement

From Figure 4.3 it is clear that the *traditional placement*, which is extremely compact, is very different from the ones built by [3] and by the algorithm proposed in this work, which are rather scattered. By comparing the placements with the distribution of the 75^{th} percentile of irradiance (Figure 4.2) it is easy to notice that both [3] and the algorithm proposed in Section 4.2.3 exploit the most irradiated areas, thus reaching a maximum improvement w.r.t. the traditional placement of 20% for [3] and of 23% for the proposed algorithm. This proves the importance of observing the evolution of environmental quantities on the roof of interest.

The differences between [3] and the algorithm proposed in this work are instead more subtle, and are mostly due to the fact that [3] ignores the correlation of irradiance evolution over time before actually placing the PV modules. The result is always an improvement of the proposed algorithm over [3], from a minimum of 1% to a maximum of 6%.

By looking at PV module placement for *roof 1* (Figure 4.3.*a*-*c*), we observe that the main variation is related to the placement of PV modules belonging to the last series (yellow). Our algorithm considers as more promising positions with a potentially lower suitable metric, but that are more highly correlated to the evolution of irradiance of the first module of the series. Also the other series differ between the two approaches, but in terms of connection of the PV modules, instead of placement. This is also explained by the fact that our algorithm gives priority to the correlation of irradiance evolution over time, rather than to the 75^{th} percentile per se, when connecting PV modules in series. This allows to minimize the bottleneck effect.

For this roof, the larger the number of PV modules, the higher the benefit w.r.t. [3] ($+1.01\%$ for $N = 32$, $+2.73\%$ for $N = 64$): the benefit is higher when the placement has to adopt not only highly irradiated areas but also more shaded positions.

For *roof 2* (Figure 4.3.d-f), we notice not only that a few panels are in different positions, but also that the the most interesting changes regards how the PV modules are connected together: PV modules number 7 and 8 in our placement (and thus belong to series 1) are moved to series 2 in the placement proposed in [3] (position 10 and 12), and are replaced by PV modules originally connected to series 2. This different connection of the PV modules allows to reach a higher advantage over [3] also in the configuration with less PV modules ($+6.06\%$ for $N = 32$). This happens because our algorithm is particularly efficient when dealing with roofs that are highly subject to shadows, like the presence of pipes and dormers (the large black area in the middle). Such encumbrances increase the impact of considering not only the typical behavior of a grid cell (i.e., the 75^{th} percentile) but also the evolution of irradiance (i.e., of shadows) over time.

Overall, it is important to note that the proposed placement algorithm, that takes into account also shadow evolution over time, improves power production not only w.r.t. traditional placements, but also w.r.t. the current state of the art [3], with an improvement in the order of kW per year even in small installations.

4.3.3 Cost analysis evaluation

The cost analysis was made by considering these parameters: $p_c = 250$, $E_p = 0.22\$/kWh$, $M_c = 15\$/kW/year$. By applying the methodology described in the previous section, we obtained the plots shown in Figure 4.4, which shows the ROI (crosses) and the PT (circles) for the two roofs when varying N (i.e., over the considered configurations). Notice that the evolution of PT is approximately the inverse of ROI (as evident from the equations).

The ROI behaviour for the first roof shows that the optimal configuration that maximizes the ROI (and minimizes PT) is not the one with the highest number of PV modules but rather the configuration with only 16 PV modules: adding more panels thus does not directly mean having better results in terms of investment, as the algorithm will continue to add panels in areas with decreasing irradiance: therefore the revenue does not grow linearly with N, as the increased power production does not compensate for higher investment and mainenance costs. A similar situation occurs for the second roof, that shows a parabolic trend: the addition of new panels can increase the ROI (i.e., decrease PT) up until the point when the ROI starts to decrease (i.e., PT starts increasing), with a maximum at $N = 40$.

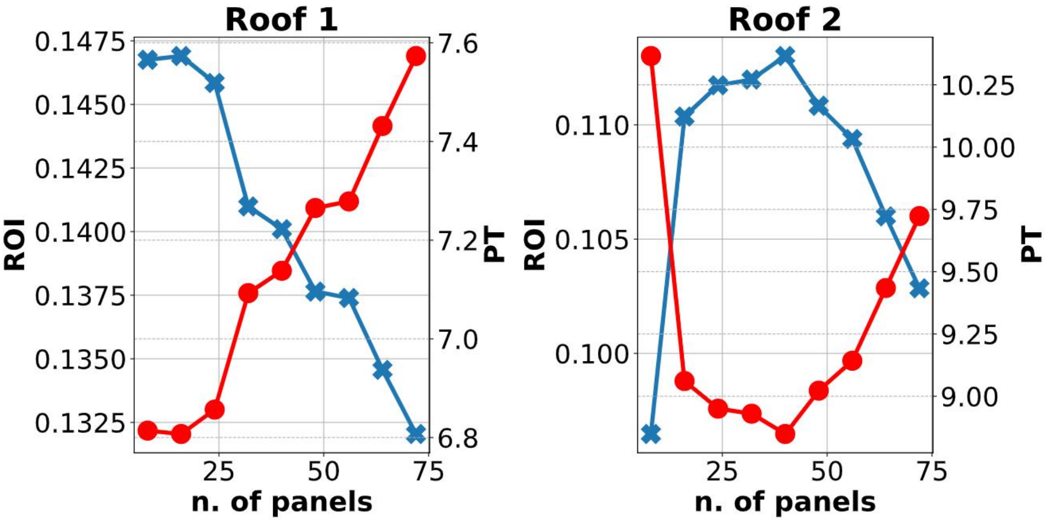


Fig. 4.4 Values of ROI (cross) and PT (circle) for roof 1 and 2 over the considered configurations (i.e., with varying N).

4.4 Conclusion

This work proposes an algorithm to find an optimal configuration for a PV installation, by observing both the cost dimension (e.g., in terms of number of PV modules and return of investment) and the power dimension (i.e., position and connection of PV modules). This allows to improve power production by up to 23% w.r.t. traditional placements and 6% w.r.t. the current state of the art [3]. Additionally, the exploration allows to derive non-intuitive optimization solutions, based on panel cost, energy price and maintenance cost, that ensure a cost-effective design of the installation and to avoid any loss of money due to a wrong sizing of the number of necessary PV modules. This shows how different scenarios require different investment strategies in order to maximize the profit of the PV investement.

Chapter 5

Design of District-level Photovoltaic Installations: an Energy Aggregator approach

5.1 Introduction

While being a promising solution, applying the prosumer paradigm to a single household is not always a viable solution: householders may not afford the cost of installation and maintenance of a PV installation, or may not be willing to make a financial investment in light of possible future earnings.

To overcome this problem, the current market solutions operate at the district level, where a number of buildings cooperate to constitute a larger PV installation and an acea aggregates the overall energy demand and takes care of selling the surplus production to the grid [100]. In this way, single prosumers do not need to care about the investment and the management of the PV systems, still achieving the advantages of potential energy independence [101].

To fully benefit from the new market paradigms, the EA must carefully design the PV installation in the area of interest, so to fully exploit its solar potential. Buildings indeed project shadows that generate heavy partial-shading effects, thus reducing the efficiency of PV power generation and requiring a careful trade-off between the size of installation (with the consequent costs) and the return of investment generated by power generation [102]: it is often the case that a larger PV installation does not lead to larger earnings, as an effect of a larger initial investment and of an ineffective power production in a portion of the installation area, subject to shading effects.

In this scenario, identifying the most suitable roofs of a district to achieve optimal PV power generation and determining the corresponding optimal PV installation is a relevant problem. Not only the problem is complex, but it also requires different skills, ranging from shadow forecast, to PV power generation and optimization, and economic estimation of the return on investment. This work proposes a solution to such a complex scenario with *a framework that works at district level to determine the optimal PV installation* from the perspective of both costs/benefit trade-off and production efficiency.

The novelty of this work lies in the following contributions:

- a GIS-based approach is used to evaluate the evolution of irradiance and temperature over the roofs of a district over one year, by achieving a good spatial resolution ($1m$) to allow an accurate estimation of the operating conditions of a possible PV installation, still operating on a wide urban area (in the order of squared kilometers);
- the identification of the optimal placements of PV modules over the district, achieved by considering the roofs of district as a whole, i.e., allowing to connect PV modules located on contiguous roofs of different buildings;
- an economic analysis to determine the payback time of the PV installation;
- a trade-off analysis that considers the payback time and the return of investment of the installation, different sizes of the PV installation and allowing different levels of PV efficiency, to determine the most suitable and the most economically convenient solution in the interest of the EA;
- the application to a district located in Turin, Italy, that will prove an improvement of power production of up to 20% and of 25% of payback time w.r.t. a traditional installation.

The chapter is organized as follows. Section 5.2 reviews relevant literature solutions. Section 5.3 presents the proposed methodology. Section 5.4 discusses the experimental results. Finally, Section 5.5 provides our concluding remarks.

5.2 Literature review

The placement and installation of PV power sources has been widely studied in the literature, with the goal of both optimizing power production and sizing, and its relation with the grid [103–105]. In this scenario, the adoption of GIS technologies is essential to enable the simulation of PV production in real urban environments, starting from either a DSM or a 3D city model of the area of interest [106–108]. The works in [109–111] analyze the potential energy production of wide areas, such entire islands or regions: they use GIS tools to extract irradiance information about the area of interest, to estimate the most promising portion for PV installation. Their analysis however does not take into account roofs, but it rather focuses on geographic areas. The works proposed in [106, 112] focus on a smaller scale, i.e., district-level, to estimate the solar potential of different roofs and calculate the expected energy production. However, both these works estimate the energy production considering a standard PV installation, without taking advantage of fine grained information to maximize such production. Other works, like [3, 113], further restrict the perspective to a single roof: they uses detailed historical data of irradiance to evaluate the best rooftop PV installation by focusing on single household installations, thus not taking the full advantage of the district-level perspective.

With respect to the literature, this work proposes a framework to investigate a relatively wide area of a urban context ($\sim 1.7km^2$ in our experimental analysis) to find a possible optimal configuration of a PV installation. The framework exploits a high resolution DSM and historical weather data to identify the most promising positions for PV installation, considering the possibility to connect PV modules located on contiguous roofs. Each explored solution is used to make an estimation of its PT to provide a tangible economic indicator for the EA, that thus can take full benefit from the installation in terms of return of investment and of generated power.

5.3 Methodology for placement on roofs of a district

The goal of the chapter is to find the best possible configuration for a PV installation for a district of a city, considering the possibility to connect PV module across contiguous roofs. The solution adopted to achieve this result is based on five main steps. The first step identifies the area of the district suitable for the installation of

PV modules. Then, we proceed with the generation of the traces of G and T for the whole area with a fine time and space resolution (A). In the second step, we evaluate a statistical measure to find which points of the suitable area are the most illuminated during the year, and thus the most promising from the perspective of power production (B). This first 2 step are described in the Subsection 3.3.1 and 3.3.2; The third step consists in the placement of PV modules to find the optimal configuration (C). Given such a placement, we evaluate the yearly production (D) and the payback time (E) to allow comparison between different solutions.

5.3.1 Optimal Placement algorithm

The third step consists of the identification of an optimal placement for the identified suitable area, given the list of PV module positions sorted by decreasing value of 75th percentile.

The algorithm starts from the position with highest 75th percentile, and goes through the sorted list to identify other positions that can be used, given the following constraints:

- no overlap with already placed PV modules;
- distance from the already placed PV modules below a threshold *maxD*;
- height difference w.r.t. already placed PV modules below a threshold *maxH*.

The constants *maxD* and *maxH* are determined by the user and they allow to connect PV modules placed on contiguous roofs or roof pitches, but ensure that the necessary cables do not generate high dispersion. This step is repeated until one series of *S* PV modules is built. Any time that a new position is chosen, it is removed from the sorted list.

Once that a series of size *S* is built, the positions excluded due to distance and height constraints are put back in the sorted list, and the algorithm starts the construction of a new series from the new position with highest 75th percentile. The algorithm ends when it is not possible to build a new series, i.e., the number of remaining positions is lower than *S*.

The resulting organization of the PV installation is made of a number of series made of *S* PV modules each, connected in parallel. It is important to note that the greedy approach adopted by the algorithm allows to connect in series PV modules with similar irradiance distribution, thus reducing the bottleneck effect caused by partial shading.

5.3.2 Power production

The last step is the evaluation of the yearly power production for the identified optimal PV installation. To achieve a measure of performance, the algorithm generates also a traditional placement of the same number of PV modules, by positioning them in the suitable area but with a more standard positioning (i.e., a compact rectangular placement that does not consider the 75th percentile of irradiance).

The yearly trace of each PV module is used to estimate the yearly power production of the overall PV installation by considering the series and parallel connections between PV modules. Given N the number of series identified in step 5.3.1 and S the number of PV modules composing each string, the resulting power of the whole installation P_{yearly} is thus derived with the following formula, reproducing the bottleneck effect mentioned in Section 3.1:

$$\begin{cases} P_{panel} &= V_{yearly} \cdot I_{yearly} \\ V_{panel} &= \min_{j=1,\ldots,N}(\sum_{i=1,\ldots,S} V_{module,ij}) \\ I_{panel} &= \sum_{j=1,\ldots,N}(\min_{i=1,\ldots,S} I_{module,ij}) \end{cases}$$

where $V_{module,ij}$ and $I_{module,ij}$ are the voltage and current extracted from the i-th PV module in the j-th string, and T is the length of the irradiance traces.

5.3.3 Economic analysis

The economic effectiveness of a PV installation is usually determined as its financial Payback-Time (PT), i.e., how much time it takes for the total savings and revenue streams to cover the total cost of the initial installation. It thus indicates the number of years of operation needed to payback the initial investment when considering also maintenance costs by using the following formula:

$$PT = \frac{I_c}{R_y - M_y}$$

Where I_c is the installation cost, R_y are the yearly revenue generate by selling all the energy produced to the grid (calculated by multiplying the yearly kWh production P_{yearly} of a configuration per E_p that is the price at which the energy is sold to the grid), M_y is the yearly maintenance cost due to the price of cleaning, monitoring, and repairing that needs to be applied periodically an efficient system.

5.4 Experimental results

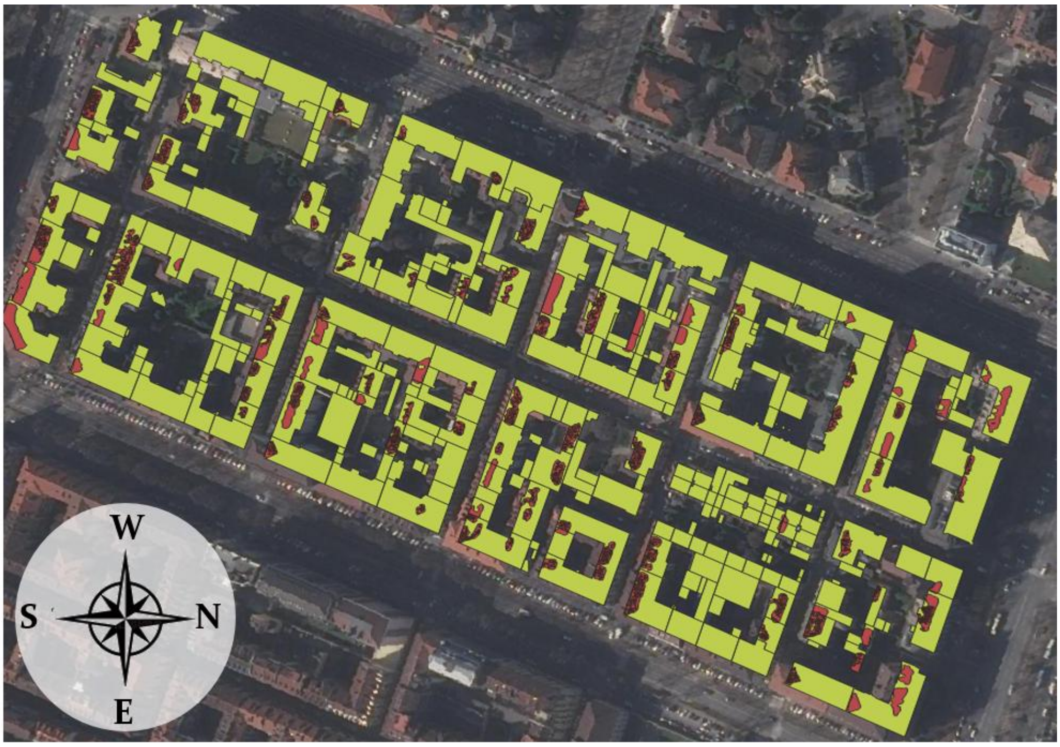


Fig. 5.1 Satellite view of the area used for the test

5.4.1 Suitable area identification and optimization

We tested the proposed framework on a district of the city of Turin represented in Figure 5.1. The satellite view of the area used for the test is overlapped with the result of step 3.3.1: the green area represents the total surface of the roofs (around $1.7km^2$), while the red area represents the area that is suitable for PV installation (about $8,340m^2$). It is interesting to notice that just 0.5% of the available roof surface is considered suitable for PV installation. This underlines that the DSM data management strategy followed in this work allows to optimize data memorization, as only few DSM points are used to filter the suitable area, and full irradiance traces are generated only for a very limited portion of the district.

5.4.2 Optimal placement setup

In our setup, we consider a PV-MF165EB3 module by Mitsubishi [2] organized in strings of $S = 8$ PV modules. The algorithm is set to allow maximum distance among PV modules on the same series $maxD = 3m$ and a maximum height difference $maxH = 0.5m$. This configuration allows to place in the same series PV modules positioned on different but almost contiguous roofs. An example of this scenario is visible in the top of Figure 5.2, that zooms on a portion of the district to show an example where the suitable area spans across two contiguous roofs (delimited by the black line in Figure 5.2).

5.4.3 Analysis of the identified PV placements

To test the performance of the placement, we executed the algorithm multiple times with different values of $minTh$, i.e., the threshold of the 75th percentile used to consider only the most promising portion of the suitable area. The results are reported in Table 5.1. When increasing $minTh$, the area considered promising is reduced as a number of suitable locations are removed as featuring a 75th percentile lower than $minTh$. Thus, both the percentage of exploited area and the number of placed PV modules decrease when increasing $minTh$. An example is shown in Figure 5.2: when $minTh$ is set to 100 W/m^2 , 21% of the available surface is considered suitable for PV placement; when $minTh$ is increased to 500 W/m^2, only 2% of the area is considered suitable for PV installation, and as a result far less PV modules (colored rectangles) are installed on the same portion of roof. If we plot the number of PV modules installed (first plot in Figure 5.3), we can observe that the decrease is not linear w.r.t. the threshold: as an example, moving from 200 to 300 W/m^2 the number of modules (and the area exploited) are reduced by $2/3$, as a wide percentage of locations has 75th percentile between 200 and 299 W/m^2. As expected, the initial installation cost decreases according to the number of PV modules installed (second plot in Figure 5.3).

Threshold $minTh$ (W/m^2)	**PV modules (#)**	**Installation area** (m^2)	**Suitable area used** (%)
100	1,792	1,540	21
200	1,536	1,319	18
300	656	561	8
400	464	394	6
500	176	148	2

Table 5.1 Percentage of area used by PV placement over the total suitable area when varying $minTh$

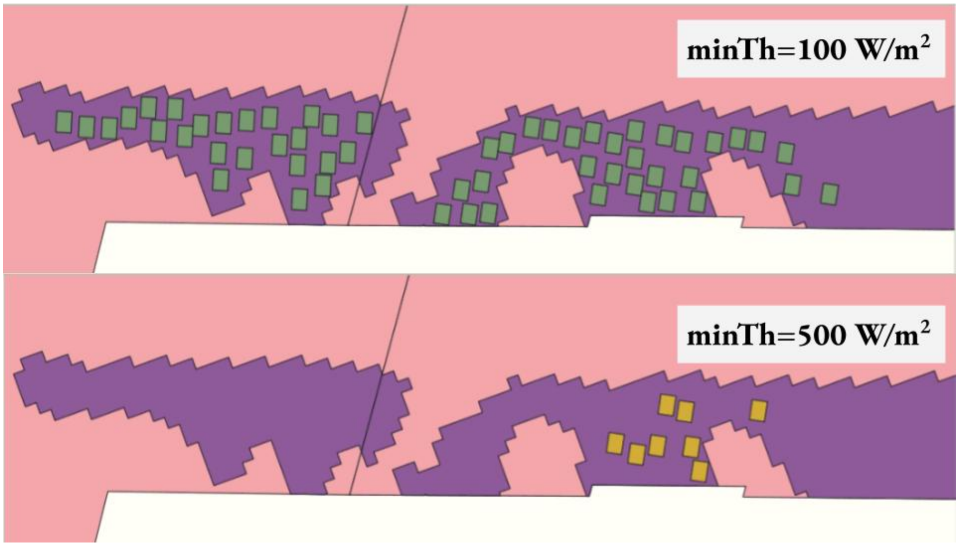


Fig. 5.2 Result of the placement algorithm on a small portion of the district (i.e., two roofs) with threshold $minTh = 100\ W/m^2$ (top) and $minTh = 500\ W/m^2$ (bottom): the pink area represents the area of the roofs, the purple area is the suitable area, and the rectangles represent PV modules placed on locations with 75th percentile higher than $minTh$. As expected, the second placement contains less PV modules, as the minimum threshold is set to a higher value.

5.4.4 Energy production performance

To analyze the performance of the proposed algorithm in terms of power outcome, we compared the yearly production of the identified optimal placements w.r.t. a traditional placement of the same number of PV modules. The traditional placement is built by installing PV modules in a more standard positioning (i.e., a compact rectangular placement that does not consider the 75th percentile of irradiance) by individually considering each roof, thus avoiding cross building deployments.

Table 5.2 shows that the production of the optimal PV installations is always larger than the one of the corresponding traditional placement, and as expected the production of the different configuration decreases linearly with the number of modules installed. However, it is interesting to notice that the improvement of power production is higher with higher values of $minTh$, with maximum improvement of 21% with threshold 500 (as shown in Table 5.2 and reported in the third plot of Figure 5.3). This behaviour can be easily explained by considering that the lower the threshold the higher the number of PV modules, and thus of potential overlap of positions of PV modules for the two placements. Vice versa, with higher thresholds the number of PV modules is reduced and the optimal placement can successfully select only the positions less affected by shading. This reduces the impact of the bottleneck effect of partial shading on the output power production of the optimal PV placement. This analysis is confirmed by the amount of area shared by the two placements, that is higher with $minTh = 100\ W/m^2$ (36%) and decreases with higher values of $minTh$, with a minimum of 15% with $minTh = 500\ W/m^2$.

5.4.5 Payback time

Using the procedure explained in 5.3.3 we evaluated the PT for both the classic and the optimal configuration considering also the different value of the threshold. The energy price considered is 0.22€ per kWh [114], the cost is 250€ per PV module and the maintenance cost considered is 15€ per PV module per year. The plot 4 in Figure 5.3 shows how the PT decreases together with the number of PV modules and that the payback times of the PV installation produced by our framework are always lower than the classical ones. In particular when the threshold is at 500 W/m^2 the configuration produced by the framework reduces by $1/4$ the PT. However by comparing the result in Table 5.2 and the Figure 5.3 we can notice while increasing

Threshold $minTh$ (W/m^2)	**PV modules (#)**	**Shared area** (%)	**Power production (MW)**		
			Optimal	**Traditional**	(%)
100	1,792	36	1,015	998	+1.6%
200	1,536	31	905	874	+3.6%
300	656	22	423	376	+12.6%
400	464	23	323	277	+16.9%
500	176	15	139	115	+20.8%

Table 5.2 Summary of the comparison among the optimal placement with varying $minTh$ w.r.t. a traditional placement of the same number of PV modules.

the amount of PV modules increases the production and therefore the earning this does not decrease the PT that instead increases. This underlines how this kind of analysis are useful for an EA that needs to take into account this economic analysis to plan his investment.

5.5 Conclusions

The chapter proposed a framework to support optimal installation of PV modules in a city district, with the goal of maximizing the profit for an EA. The approach is based on an efficient management of DSM data, that generates detailed irradiance traces only for the promising portion of the district roofs ($\sim$ 0.5% of total district area). The data is then used to build an optimal placement of PV modules, that can be parametrized to exclude positions affected by shading and by a discontinuous irradiance over time. The determined placement allows to find the suitable trade-off between initial investment, power production and payback time of the installation, and proved to generate a surplus power production of up to +20% w.r.t. a traditional installation.

As future works, we plan to extend the proposed solution by including new constraints, such as i) limiting the maximum number of PV modules to install according to the user's budget and ii) limiting the PT to a maximum value defined by the user.

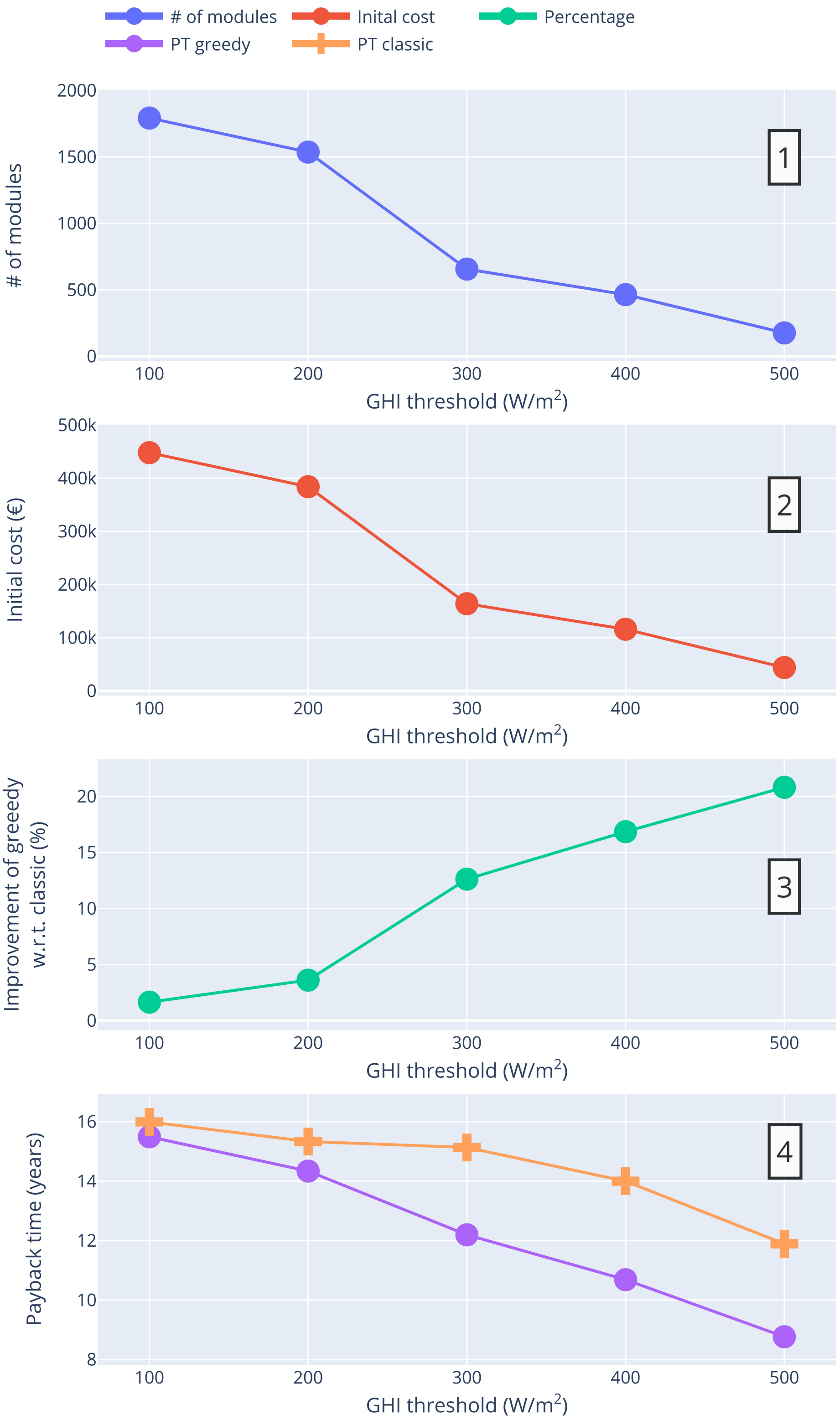


Fig. 5.3 Behavior of the placement algorithm with different values of $minTh$: number of PV modules (1), initial installation cost (2), improvement of power production w.r.t. the traditional placement (3), payback time (4) (purple for the proposed algorithm, orange for the traditional placement).

Chapter 6

A framework for economic and environmental benefit through Renewable Energy Community

6.1 Introduction

Nowadays, RESs provide approximately 14% of the global energy demand, and this percentage is estimated to rise to 50% in 2040-2050 [115]. This shift toward RES is pushed forward by the urgent necessity to reduce the global level of CO_2 emissions [116] through (i) international agreements [117] and the adoption of *carbon-taxes*, used by more and more countries to discourage the usage of fossil fuels [118], and (ii) the application of incentives for the deployment of RES systems, which have proved to be extremely effective in several countries [119].

In this context, PhotoVoltaic (PV) energy has a primary role as it is known to be a sustainable, cost-effective, reduced footprint source of energy that requires minimum maintenance [120]. Projections foresee that PV energy will provide 25% of the global power generation by 2040-2050 [121]. This success will be partially enabled by the fact that, compared with other renewable energy sources such as wind turbines, PV systems are also suited for urban and industrial environments [122].

The adoption of PV systems is also fostered by the diffusion of the *prosumer* paradigm, i.e., a new kind of energy market player that can both produce and consume energy. In the future, a prosumer will be able to use a PV system (or any other RES) not only to produce energy for self-consumption but also to sell production surplus [123]. Such a solution seems promising, thanks to affordable costs and government incentives. However, in the case of a single household, the initial investment and the economic effort needed to maintain the entire system may discourage such an environmental-friendly choice [124].

A possible solution to the above challenges is assessed by the emerging *Renewable Energy Community* (REC) paradigm, defined by the International Renewable Energy Agency as a group of citizens that work together to reduce their environmental and economic impact [125]. The key feature of a REC is “resource sharing”, i.e., the idea of sharing PV modules and rooftops to bring benefits to the whole community [126]. RECs can facilitate the diffusion of renewable energy sources and encourage investments. At the same time, they can provide direct benefits to the members in terms of smaller energy costs and better energy efficiency, as proved by several initial studies [126–128].

RECs need detailed planning to achieve their main objectives, that range from obtaining profits for the stakeholders to acquiring equal benefits for all participants [129]. Selecting the correct REC schema depends on both demographic and geographic information. First of all, communities located in an urban environment may have limited space for the deployment of PV panels. Secondly, the latitude where the community is located deeply affects day length and, therefore, the quantity of energy produced. In addition, the demographic profile of the community delineates the budget for the investment, the expected savings or profits, and the projected required energy. As a consequence, it is of paramount importance to predict the impact of RES installations.

In this perspective, our goal is to facilitate the installation of RES. Focusing on PV technologies, we present an automatic software framework that exploits available geographic, demographic, historical data, and models of power generation to estimate the economic and pollution impact of the installation and to maximize the benefit for the REC.

Built upon our previous works [130, 131], the framework includes the four main stages outlined in Figure 6.1:

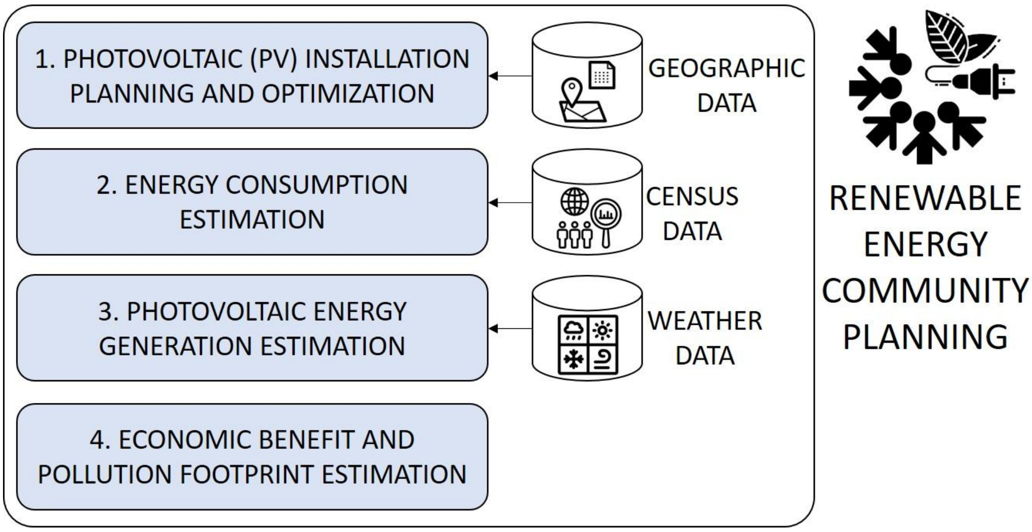

1. PHOTOVOLTAIC (PV) INSTALLATION PLANNING AND OPTIMIZATION
2. ENERGY CONSUMPTION ESTIMATION
3. PHOTOVOLTAIC ENERGY GENERATION ESTIMATION
4. ECONOMIC BENEFIT AND POLLUTION FOOTPRINT ESTIMATION
GEOGRAPHIC DATA
CENSUS DATA
WEATHER DATA
RENEWABLE ENERGY COMMUNITY PLANNING

Fig. 6.1 Our solution to support the planning of RECs; we (1) exploit geographic information for planning district-level PV installations, (2) measure the energy power consumption for the district of interest from census data, (3) estimate the resulting yearly PV power generation from weather data, and (4) evaluate the economic benefit and pollution footprint to allow the REC to make informed decisions.

1. *PV installation planning and optimization:* The proposed framework automatically designs the most suitable PV installation, given the geographic constraints of the area of interest (modeled as GIS data) and historical data of irradiance. The algorithm explores different configurations to maximize irradiance exposure and power production, and it allows to place PV modules across different rooftops, as part of the REC sharing approach.

2. *Energy consumption estimation:* The power demand of the REC is estimated by considering the demographic information of the district. Census information is automatically elaborated to derive disaggregated power consumption traces reflecting the district inhabitants (e.g., in terms of family size and behavior).

3. *Estimation of power generation:* The output of the installation is obtained from power generation models of the PV modules. These are obtained from yearly traces of irradiance by taking into account the analysis of phenomena such as shading on the distribution of irradiance over the district.

4. *Estimation of the benefit for the REC:* In terms of (i) economic benefit, due to a reduction of energy demand from the grid thanks to self-production,

and (ii) reduction of the pollution footprint, measured as a reduction of CO_2 emissions [132].

6.2 Identification of possible REC

The transformation into prosumer or the participation to a REC could always be beneficial for the customer, which in general can obtain some economic benefit. In the case of PV systems, however, the actual benefits heavily depend on the number of panels the user can install. For example, in rural areas, the user often lives in a single-family house and therefore he can use all the power produced by its panels. On the other hand in urban contexts, customers often live in apartments located in tall buildings, in this case, the energy produced by the panels has to be shared among many inhabitants. This poses the challenge to find a method to divide areas where the creation of a REC is feasible for its participants from areas where other strategies needs to be applied (for example using the EA approach) For this work, we decided to introduce a naive but effective parameter that can help to do so. The main idea behind this parameter, shown in the Equation 6.1 is to evaluate the ratio between the areas of the roofs an the number of inhabitants of a given zone

$$k = \frac{\text{Area of roofs}}{\text{number of inhabitants}} \tag{6.1}$$

In this work, we calculated this parameter for each of the census area of the city of Turin by using publicly available demographic data [133] to obtain the number of inhabitants of each zone, while we evaluated the surface of the roofs from the result obtained by the procedure described in 3.3.1. The result of this operation can be seen in Figure 6.2 where lighter areas indicate higher values of k.

This preliminary selection helped us to individuate the most promising area for the creation of a REC, therefore we selected five districts in and around the city of Turin, a large city in northwest Italy. These were chosen in order to test the proposed framework in different contexts, representing a wide variety of urban and suburban scenarios (in terms of population size and distinct building densities). It is however important to note that *this work does not propose a country-specific solution*. Indeed, the proposed framework is applicable to any geographic area with minor

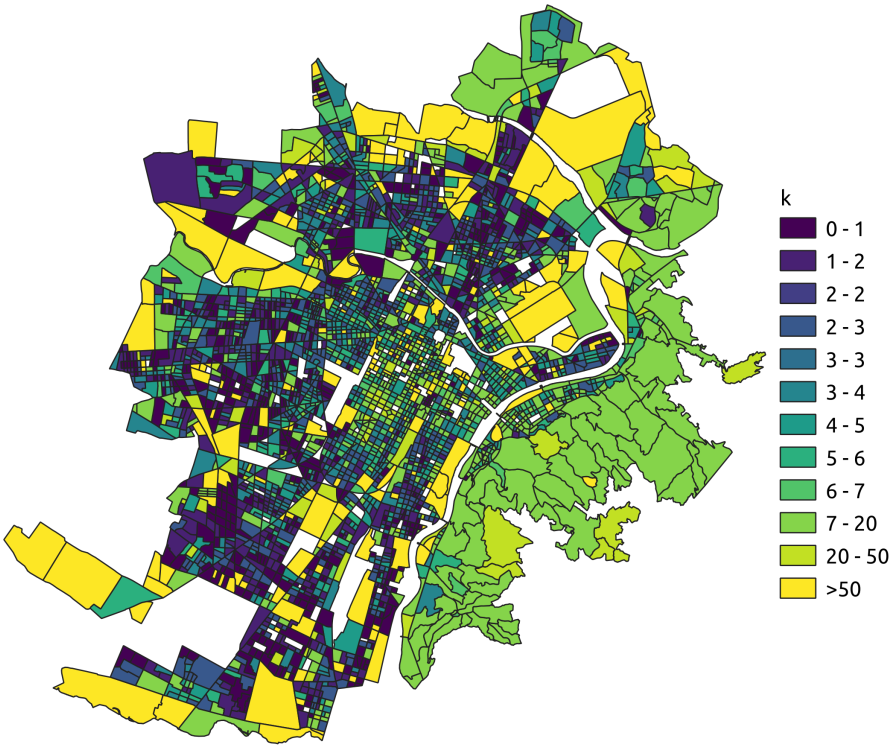


Fig. 6.2 Visualization of the value of parameter *k* calculated with 6.1 for the census zone of the city of Turin

configuration changes (for example, required to leverage different latitudes [77, 134]) once that the necessary information is available.

This chapter is organized as follows. Section 6.3 describes related works in literature and the main differences with our previous works [130, 131]. Section 6.4 introduces our framework, and Section 6.5 applies it to the case studies. inally, Section 6.6 reports some concluding remarks.

6.3 Literature review

6.3.1 RECs and PV installations

RECs play a crucial role in successful energy transitions as they increase the flexibility and the resilience of energy systems, and spread the acceptance of bigger renewable energy projects [18]. RECs are composed of a variety of actors (such as households, public institutions, business activities, etc.) sharing the common goal of producing energy from renewable sources to maximize self-consumption, hence reducing the need to withdraw energy from the electric grid.

However, a simple installation of a shared renewable energy plant through an investment of a community is not enough to set up a REC. Many other challenges need to be taken into account, such as legal aspects, grid connections, demand flexibility policies, and storage systems [19]. The legal institution of a REC is the first challenge that needs to be analyzed since regulations vary from place to place. For example, EU and North America have distinct policies [20, 21] with differences even among countries [22]. In addition to that, due to the complexity of the system and the variety of actors involved, RECs need careful planning to ensure efficiency and success [25]. The first thing to consider is the type of renewable energy that should be deployed. As previously mentioned, cost reduction makes PV systems the perfect choice for a collaborative approach; thus, the literature provides several solutions to plan a PV system in a REC [135] or with Demand Response Policies [136, 137]. This work however goes a step further as it focuses on the *optimization of the PV placement in the context of a REC*, providing a more efficient cross-rooftops configuration of the whole system with respect to traditional individual rooftop installations.

6.3.2 Optimal planning of PV installations

Finding optimum arrangements for RES is of paramount importance, and it has been analyzed from several perspectives in the literature.

Some works focus on the combination of different types of RESs and analyze their impact on the resilience of the electrical network [103, 104, 138, 139].

Other works concentrate on the organization of the panels. GIS technologies, such as DSM and other 3D modeling techniques, are used to analyze real scenarios allowing digital representations of the area under study [106–108]. Some works use GISs information to improve their estimates of the power production of PV systems. Other authors [109–111] use GISs to estimate power production in large areas (such as an entire region or an island) using low-resolution maps.

Sun et al. [141] use a similar approach to estimate the PV potential at a regional level, analyzing costs and benefits, and evaluating the reduction of CO_2 emissions. Anyway, their approach does not fully exploit GISs as it does not suggest any strategy to maximize the production of a PV system.

Raul et al. [142] propose a solution to evaluate PV production at the national level. They include in their work an economic analysis taking into consideration the different socioeconomic situations across the country. They use cadastral data and apply a correction factor to estimate the actual surface available for each PV modules. Unfortunately, they do not consider the real topology of the rooftops and therefore they do not provide any configuration for the PV module.

Other works [106, 112, 143] analyze smaller geographical areas to evaluate power production of PV systems. However, only traditional panels installation are considered, and therefore, production is not optimized.

Jacques et al. [144] use high-resolution Light Detection And Ranging (LiDAR) images to identify the rooftops on which installing the PV modules. However, they evaluate the power production without taking into account the exact position of the modules as they consider the entire rooftop as available.

Bergamasco et al. [111] take into consideration the rooftops of an entire city and use the GIS to select with high accuracy the rooftops available for the PV modules. Anyway, they do not give any indication regarding the actual placement of the

	PV potential	**Available surface**	**Historical data**	**PV module specifica-tion**	**15 min resolution**	**Optimized placement**	**Load profile analysis**	**REC approach**	**Energy Savings**	**CO_2 emissions**
Baranay et al.[109]	✓	x	x	x	x	x	x	x	x	x
Khemiri at al.[110]	✓	x	x	x	x	x	x	x	x	x
Bergamasco et al.[140]	✓	x	x	x	x	x	x	x	x	x
Sun et al.[141]	✓	x	x	x	x	x	x	x	x	✓
Miranda et al.[142]	✓	x	x	x	x	x	x	x	✓	x
Kucuksari et al.[106]	✓	✓	✓	✓	x	x	x	x	x	x
Bracco et al.[112]	✓	✓	✓	x	x	x	x	x	✓	x
Huang et al.[143]	✓	✓	✓	x	x	x	✓	x	✓	x
Jacques et al.[144]	✓	✓	x	x	x	x	x	x	x	x
Bergamasco et al.[111]	✓	✓	✓	x	x	x	x	x	x	x
Damiri et al.[145]	✓	✓	✓	✓	x	x	✓	x	✓	x
Vinco et al.[146]	✓	✓	✓	✓	✓	✓	x	x	x	x
El et al.[113]	✓	✓	✓	✓	✓	✓	✓	x	x	x
Cielo et al.[147]	✓	✓	✓	✓	✓	x	✓	✓	✓	x
Syed et al .[148]	✓	✓	✓	✓	✓	x	✓	✓	✓	x
Our previous works [130, 131]	✓	✓	✓	✓	✓	✓	x	x	✓	x
Proposed	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 6.1 A synoptic comparison between our framework and the more similar works recently presented in the literature.

module, and they provide a rough estimate of the power production considering the azimuth angle and the entire surface.

Another weakness of all previous works is that they do not use historical data to estimate energy production. However, it has been demonstrated that this information can be extremely effective in improving the accuracy for those approaches that use only data obtained through theoretical equations [149]. For such a reason, several studies use hourly or quarter-hourly historical data to provide more precise estimations with realistic sky conditions.

Damiri et al. [145] use meteorological and solar irradiance data to estimate the power production of a PV configuration placed over an industrial rooftop. However, even if they provide a feasible PV module configuration, they consider only a single rooftop.

Similar studies, like [146, 113], exploit historical data to identify the best placement for the PV modules over different buildings. However, they do not focus on RECs, and rooftops are considered individually and not as a shared resource.

All previously mentioned works consider a single building or household and not a REC, where different buildings can be organized as a single shared resource. Sharing rooftops allows cross-rooftop connections and, more in general, energy sharing, with enormous benefits in terms of power production [150, 151]. For this reason, Cielo et al. [147] use historical irradiation and disaggregated consumption data to evaluate the impact of shared PV systems on a community. However, irradiation is not used to optimize PV module placement and maximize production.

Syed et al. [148] evaluate the benefit of a PV system shared among households. The authors also compare household load profiles with energy production to evaluate the possible advantages of such a shared system. However, even in this case, the panels' position is not optimized to maximize energy production, and there is no analysis on the reduction of CO_2 emissions.

Overall, the novelties of our approach with respect to the previously mentioned works are summarized in Table 6.1.

The framework proposed in this work is based on [130, 131] from which it derives the procedure related to evaluating the available surface and the model used to assess the energy production of a PV module. This framework, however, adds some major novelties and improvements with respect to our previous works [130, 131].

It introduces the possibility of testing PV configuration with modules arranged in different directions. This feature increases the optimization procedure's flexibility to maximize the system's production further. Moreover, we abandoned the previous approach used to estimate the economic benefits that were too tightly related to the cost of energy, which may widely differ from one place to another. Therefore in this work, we evaluated such benefits as the percentage of self-consumption since this gives a more general but still valid indication of the advantages obtained from participating in a REC. This approach required an additional step in the processing pipeline that uses census data and smart-metering measurements to provide accurate energy demand profiles for the REC under study. We analyze five different areas covering three different REC scenarios spanning from urban to rural (whose size ranges from about 6000 m^2 to about 144000 m^2) to optimize the PV system. We selected these areas to analyze scenarios with different characteristics, such as population and building height or distances. To summarize the primary novelties are:

- The improvement of the cross-rooftops algorithm to find the suitable area for PV system installations by increasing the search space of the optimization problem under analysis.
- The reduction of the dependency of the economic benefit on the fluctuation of the prices of hardware and energy.
- The integration of new studies, which integrate both actual census data and real world household load consumption.
- The estimation of the environmental impact in terms of CO2 emissions, which decrease by creating new RECs .
- The analysis of different case studies to analyze the possible impact of REC in various urban and rural scenarios"

Our framework is built upon and extends our previous works [130, 131]. It can find the available surface in roofs to install PV systems. It also adds new features such as the load profile analysis, the REC approach, and the evaluation of the CO_2 emission. The REC approach considers roofs as shared resources and allows cross-roof PV connections. We analyze five different areas (whose size ranges from about 6000 m^2 to about 144000 m^2) to optimize the PV system. Our framework uses fine-grained DSM together with historical weather data (with a resolution of 15 minutes) to provide an optimal configuration of the PV panels placement.

The PV module we consider can be easily changed with other models to ensure a more realistic estimation and increase the flexibility of our framework.

We assume all rooftops as belonging to a shared resource and we connect panels located on different buildings (i.e., we allow cross-roof connections). We prove that our estimated power production is always more significant than the one generated with an equivalent traditional installation. We compare the estimated power production with the aggregated power demand to evaluate the self-sufficiency capabilities of the REC. We estimate the reduction of pollutants (in terms of CO_2 emission) for the scenario without the REC.

6.4 Methodology for placement in REC context

Our main target is to build a framework to analyze the economic and pollution benefits that RECs can obtain by sharing PV installations. From a high-level point of view, our framework includes three main phases containing the six different steps represented by the numbered blocks in Figure 6.3:

1. In the first module, the framework uses DSM data to extract the surfaces of the rooftops on which it is possible to install PV modules. This analysis is then used to extrapolate the temporal evolution of irradiance and temperature for the area under study and it is described in the Subsection 3.3.1

2. The second module performs a statistical analysis of the data generated during the first stage to find the portion of the rooftops with the best irradiation condition (which is obviously expected to produce more power than poorly irradiated ones). This procedure is described in the Subsection 3.3.2

3. In the third module, the previous evaluations are used to find an optimal configuration for the modules of the PV system.

4. Such a configuration is then used by the fourth module to estimate the power production of the PV system with a procedure that uses DSM, meteorological data, and the datasheet of the PV cell.

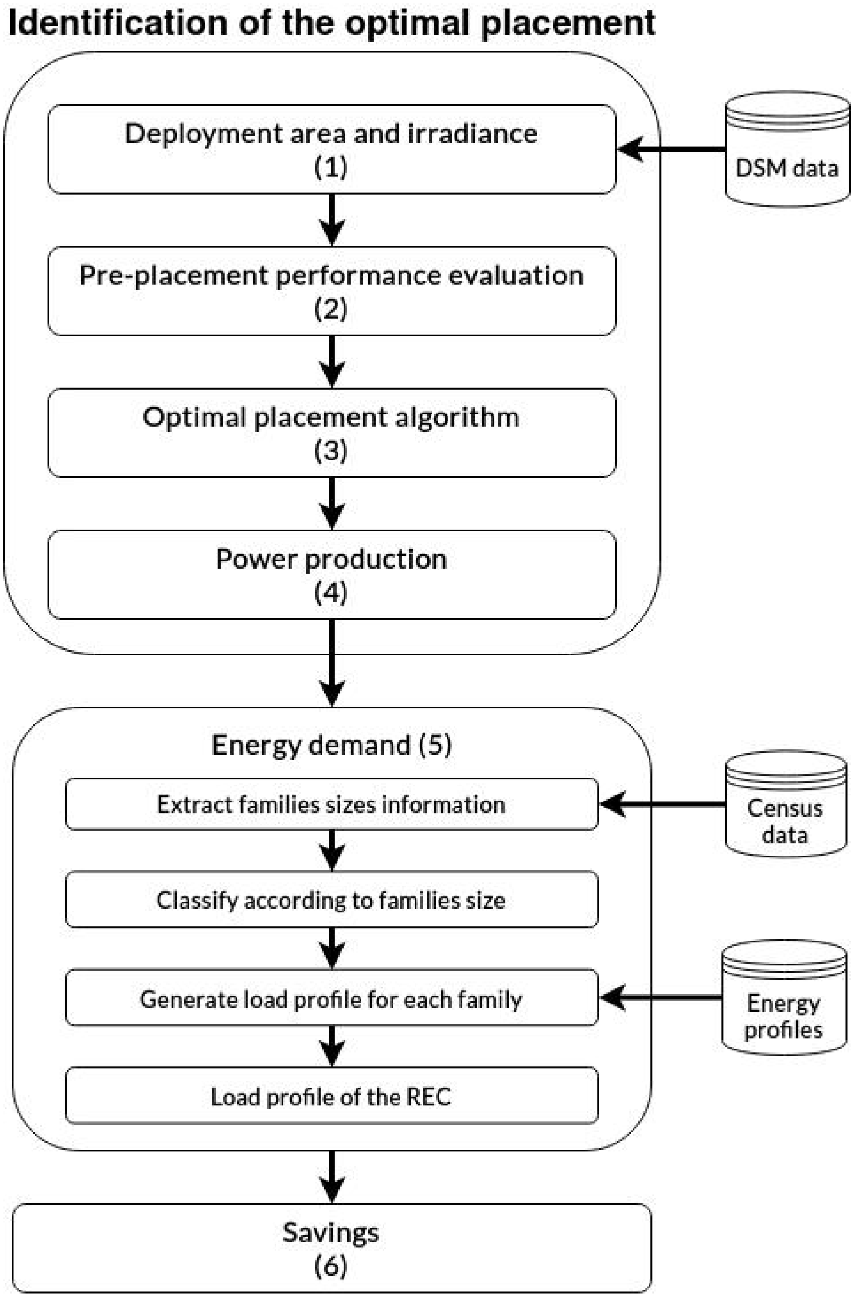

Identification of the optimal placement
Deployment area and irradiance (1)
DSM data
Pre-placement performance evaluation (2)
Optimal placement algorithm (3)
Power production (4)
Energy demand (5)
Extract families sizes information
Census data
Classify according to families size
Generate load profile for each family
Energy profiles
Load profile of the REC
Savings (6)

Fig. 6.3 Diagrammatic representation of our framework with its three main phases and six main pipeline steps.

5. The fifth module estimates the power demand of each partner of the community.

6. In the sixth and last module, the framework uses our estimates (in terms of power generation and power requests) to evaluate the potential savings in terms of both energy and CO_2 emissions.

6.4.1 Optimal placement algorithm

The third module (block (3) in Figure 6.3) identifies the best possible panel configurations selecting them from the sorted list generated by the previous module.

Our procedure selects the first, and most promising, configuration. Then, it runs through all remaining configurations to select the ones compatible with the selected one according to the following constraints:

1. Each new PV module should not overlap with previously placed modules.
2. The horizontal distance between two consecutive modules should not be larger than a threshold *maxD*.
3. The vertical distance between two modules should not be larger than a threshold *maxH*.

The two thresholds (*maxD* and *maxH*) are used to connect modules that are closed enough (horizontally and vertically, respectively) and to consider all rooftops of the community as a shared resource.

We iterate through the process until we obtain a series of *S* panels. Every time we select a configuration, we remove it from the list of the remaining ones. When we complete a series, we re-insert all positions that did not respect constraints (2) and (3) in the sorted list, whereas if a position did not respect constraint (1) we completely remove it from the list. We terminate the process when the number of available positions is less than *S*, as this condition implies that it is not possible to form a new complete series of PV modules.

The result of this procedure is a *group*, i.e., a set of PV modules placed on contiguous rooftops in which modules are first connected in series and then series of modules are connected in parallel. Due to the structure of the algorithm, PV modules that belong to the same series are likely to have a similar evolution of radiance over time. This avoids bottleneck effects caused by the partial shading of one panel of the series.

Notice that, in our experimental evaluation, this procedure is executed three times for each case study to consider three different panel orientations, i.e., the ones with only vertical panels, the ones with only horizontal panels, and the ones with both vertical and horizontal panels. The evaluation of different orientations is an important novelty with respect to previous works (such as [130, 131]) as our algorithm may place panels belonging to the same series not side-by-side as it would happen in traditional PV configurations.

6.4.2 Power production

The fourth module (block (4) in Figure 6.3) estimates the yearly power produced by the identified configuration. During the first step, using the approach described in [152], this module evaluates the yearly traces of voltage and current for each PV module by extracting relevant information from datasheets of real PV modules. Then, it computes the overall power production, taking into consideration all PV modules and their connections (as defined in Section 3.1): The current generated by a series of PV modules is limited by the module producing the minimum current, and the voltage generated by the entire PV installation is constrained by the series producing the minimum voltage. This methodology increases the flexibility of our framework since we can easily evaluate the power production obtained using PV modules with different characteristics.

6.4.3 Energy demand

This module estimates the hourly power demand of the community and compares it with energy production. The logic of this phase is reported directly inside the block (5) of Figure 6.3.

To estimate the power requirements, we use a real dataset provided by Midori s.r.l. [153]. This dataset contains the energy consumption profiles of more than 90 houses located in Turin and disaggregated at the appliance level. The dataset is pre-processed using the Jenks Natural Breaks method [154] to classify consumption data according to the number of inhabitants of each house. Using these results, the module can assign to each house a realistic power demand profile by selecting one of the houses with the same number of inhabitants. The results of this preliminary classification are combined with the census data [133] to build a realistic virtual population for the geographical area of interest. The resulting synthetic population is extremely realistic and presents different energy consumption behaviors tailored to the demographic composition of the area. The different profiles are finally combined to obtain a realistic profile for the entire community for one year.

6.4.4 Savings

In the last module (block (6) in Figure 6.3), we compare the power demand of the community with the power produced by its shared PV system to establish the degree of energy self-sufficiency and the reduction in CO_2 emissions. We compute the percentage of self-produced energy as the ratio between the yearly energy demand of the REC and the yearly energy production of the shared PV system. Moreover, to estimate the reduction in terms of CO_2 emission, we compare the case in which the yearly energy demand is satisfied only by fossil fuel energy sources with the case in which a portion of this energy is self-produced directly by the REC. In this case, we also consider the small but not negligible amount of CO_2 generated by all PV modules. CO_2 are computed following the Equation 6.2.

$$CO_2 = \frac{(PV\,energy)\ \cdot\ (PV\,CO_2\,emissions)}{(Total\,energy\,demand)\ \cdot\ (fossil\,fuel\,CO_2\,emissions)} \tag{6.2}$$

6.5 Experimental results

We test our framework on different neighborhoods in the city of Turin to verify our framework for a large variety of scenarios:

- *Town Center*: Area of a city with a high building density where buildings may have different heights, ranging from 3 to 7 or more floors. The age of the structures varies a lot, as we have both historic and modern buildings close to each other.
- *Outskirts*: Area with a medium density of buildings with very similar heights (from 1 up to 3 floors). Usually, buildings are populated by one or at most two families, and backyards often separate them. This category includes also villages.

- *Suburban area*: Area with small houses located very sparsely on the territory, each owned by a single family.

We run 5 experiments: Three located downtown, one in the outskirts, and one in the suburban area. We ran the proposed optimization procedure on a desktop equipped with Ubuntu 20.04, a CPU Intel(R) Core(TM) i7-8700 @ 3.20GHz, and 16 GBytes of RAM. We needed 22 minutes to obtain an optimal solution for all the case studies proposed in this work. The first objective of these tests is to prove that our framework builds PV installations that generate more energy than the ones placed by traditional methods. Our secondary objective is to prove that our configurations reduce both the energy withdrawn from the power grid and greenhouse gas emissions. For each outline, Table 6.2 reports the size of the portion of rooftops that could be used to deploy PV modules (i.e., the areas facing south and without encumbrances). Figure 6.4 shows the result obtained by our panel allocation algorithm for the first scenario. We highlight the entire surface in green and the selected area in red. It is worth noticing how the area suitable for the installations is just a small portion of the rooftop surface and ranges from 12.4% up to almost 25%. We obtain similar results for all outlines, even with the ones including different building profiles, such as the Town Center Area 2 and the Outskirts Area. Moreover, the buildings of the Suburban Area cover three times the surface available in the Outskirts, but the useful surface is just twice as large as the one identified for the Outskirts. This information can be extremely helpful to identify the areas that could benefit more from a PV system.

Case study	Buildings area (m^2)	Suitable area (m^2)	percentage
Town center 1	5,702	909	15.9
Town center 2	4,032	808	20.0
Town center 3	9,275	1,553	16.7
Outskirts	3,233	804	24.9
Suburban	10,232	1,268	12.4

Table 6.2 Comparison between the total area of all rooftops and the area suitable for the installation for each one of the 5 case studies.

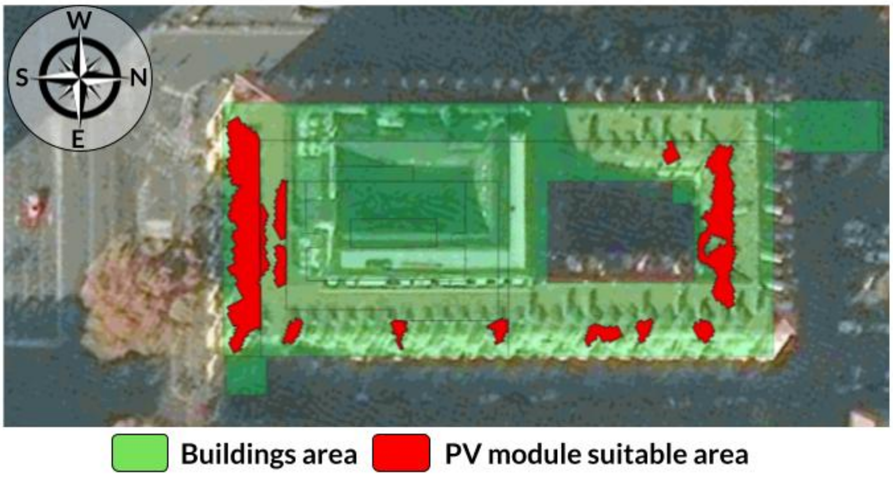


Fig. 6.4 A satellite image representing some real cross-building rooftops (highlighted in green color) with the area actually exploited to install PV modules (represented in red color).

The next module of the framework identifies the optimal positions for the panels while respecting the constraints described in Section 6.4.1. For this test, we take into consideration PV modules of type MF165EB3 produced by Mitsubishi [2], and we consider a series of *S*=8 PV modules. We set a maximum horizontal distance between adjacent panels of *maxD*=3 meters, and a maximum vertical difference of *maxH*=0.5 meters. Figure 6.5 shows some examples of PV configurations generated by our placement algorithm considering several adjacent rooftops. We can notice the different orientations considered by the algorithm, i.e., PV are modules placed only vertically (top), only horizontally (center), and both vertically and horizontally (bottom picture).

To better evaluate the performance of our placement algorithm, we compare its results with the ones delivered by a traditional algorithm. Thus, for each configuration (i.e., for each possible orientation considered by our algorithm), we generate a conventional layout with the same amount of PV modules placed on the same rooftop and orientation. This process ensures that an increased installed capacity does not determine a higher power production than a traditional placement. Figure 6.6 compares the energy production of the different layouts for each district. It is possible to observe that in the first scenario the power productions of the different placement algorithms are very similar, whereas in the other situations the greedy algorithm always outperforms the traditional one. In particular, when the greedy algorithm

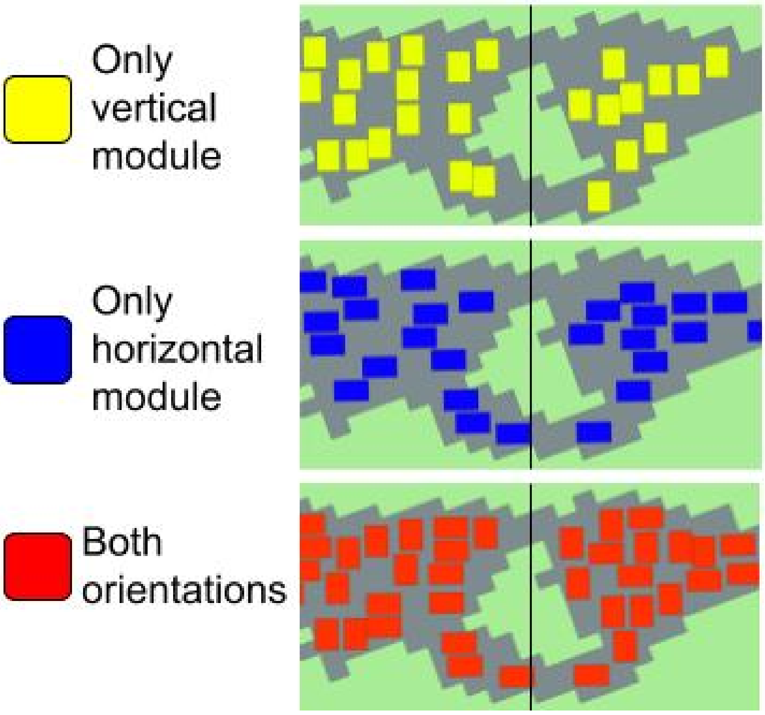


Fig. 6.5 Outcome of our placement algorithm: PV modules can be placed onto contiguous rooftops and with different orientations.

considers both vertical and horizontal orientations, we obtain the best results, with improvements that vary from 15% to 60%.

As the panel configurations that guarantee the best power production are the ones generated by the greedy algorithm and allowing panels with both orientations, we verify which percentage of the energy demand could be satisfied by these layouts. For each community under test, we compare the total energy consumption (obtained with the method introduced in Section 6.4.4) with the correspondent production of the panels. Results are shown in Figure 6.7. For each one of the scenarios analyzed, the plots represent the difference between the generated energy and the consumed energy. When the production is larger than the consumption the values are represented in green color; plots are in red, otherwise. It is easy to notice that, only for some of the layouts and only for specific periods over the year, the production of the installation can completely satisfy (or even exceed) the energy demand. In particular, the best balance is obtained during the daytime of the summer for both the Town Center Area 2 and 3, and partially for the Outskirts Area.

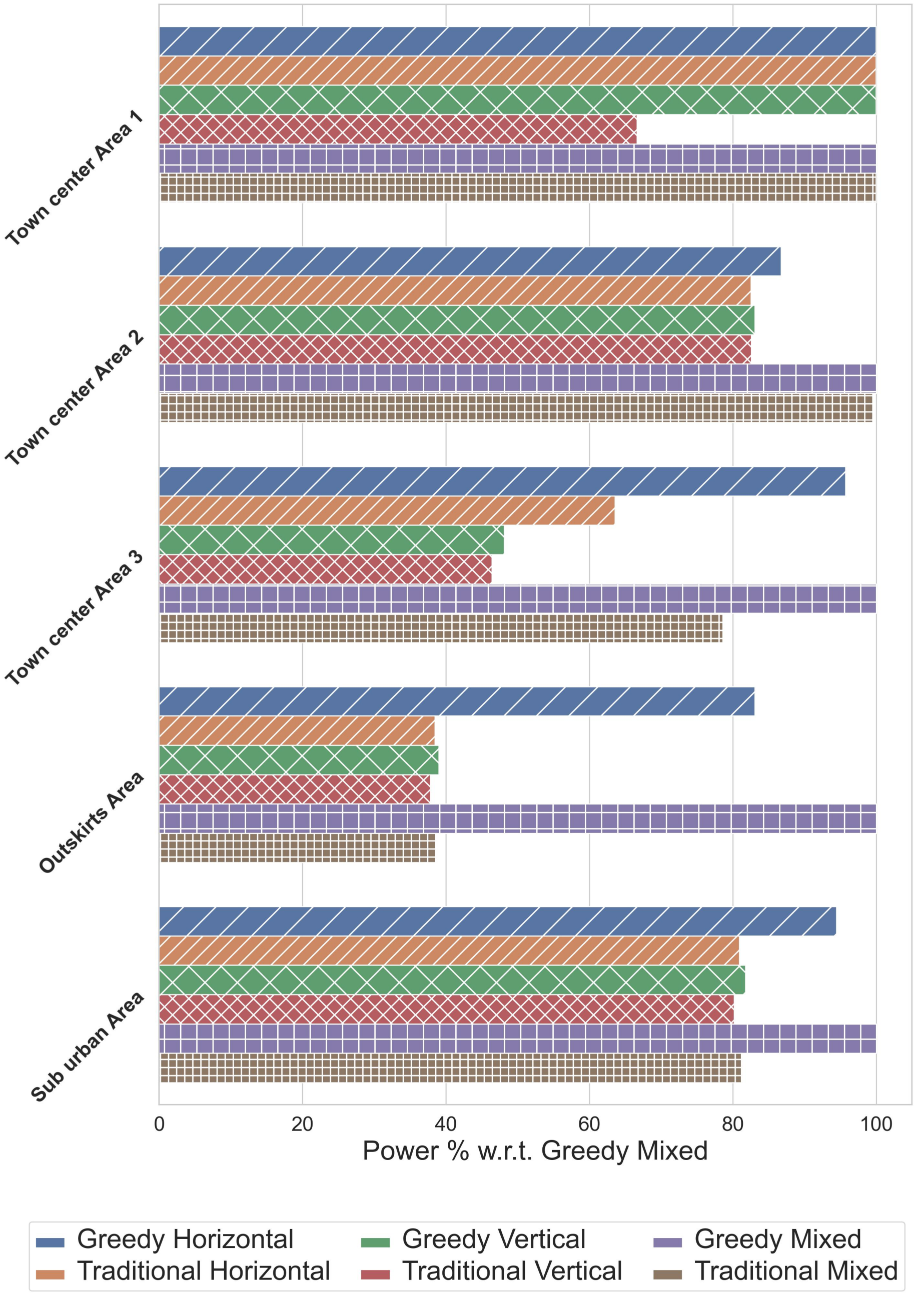


Fig. 6.6 Power production of the different placing algorithms for the different areas. Power production is normalized with respect to the solution proposed in this work, i.e., using a greedy placement and both vertical and horizontal orientation of PV modules.

However, even if RECs do not become self-sufficient, our PV installations are still beneficial. We evaluated the economic benefit obtained by REC as the percentage of

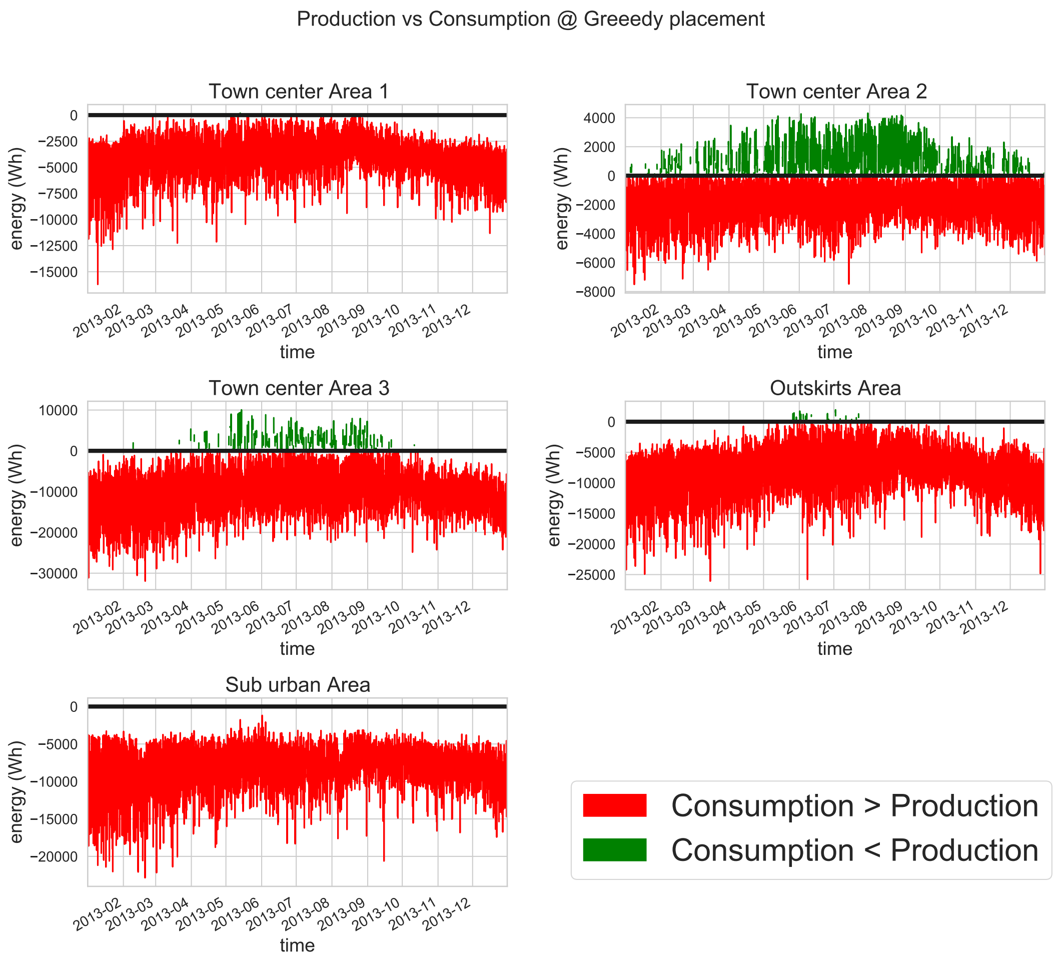


Fig. 6.7 Each graphic plots the difference between the energy production and the energy demand of the community. The y axis represents the power: The green color is used when the production is larger than the demand. The x-axis represents time, considering days as a time unit and one entire year.

self-consumption the PV system can provide. We indicated this value in Figure 6.8, which shows that our PV settings may reduce the amount of energy withdrawn from the power grid by a percentage varying between 6% to 45% over a year. We can also consider those values as the percentage saving of the REC participant on the energy bills. In addition to this, PV systems may have a very positive impact on pollution. Figure 6.9 shows that they can reduce up to 35% yearly CO_2 emissions. In this plot, we take into account the average amount of CO_2 emission due to the electricity generation derived from fossil fuels in Italy [155] and also the small (but still relevant) amount of emission related to the generation of PV panels. Our results show that, in the best case, the amount of CO_2 saved corresponds to the yearly emission of almost 1,180 cars [156].

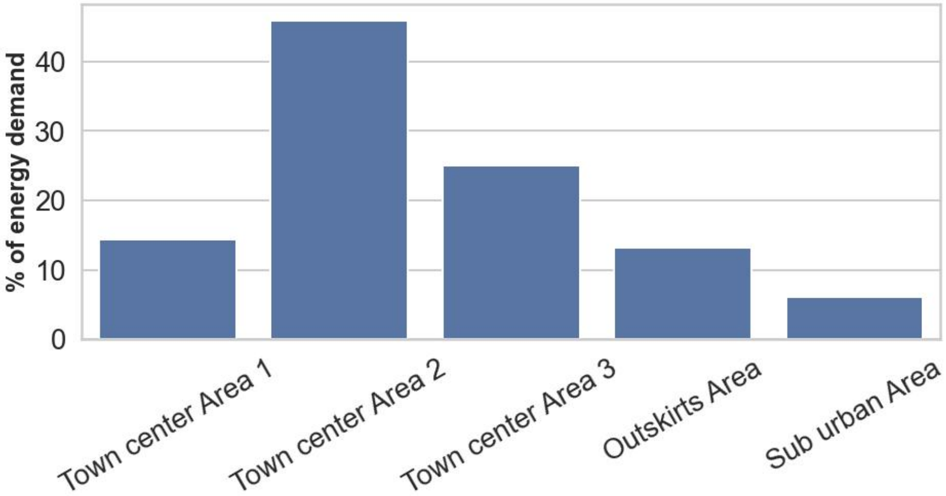


Fig. 6.8 Percentage of the energy demand that can be satisfied by a shared system over a year for each area under analysis.

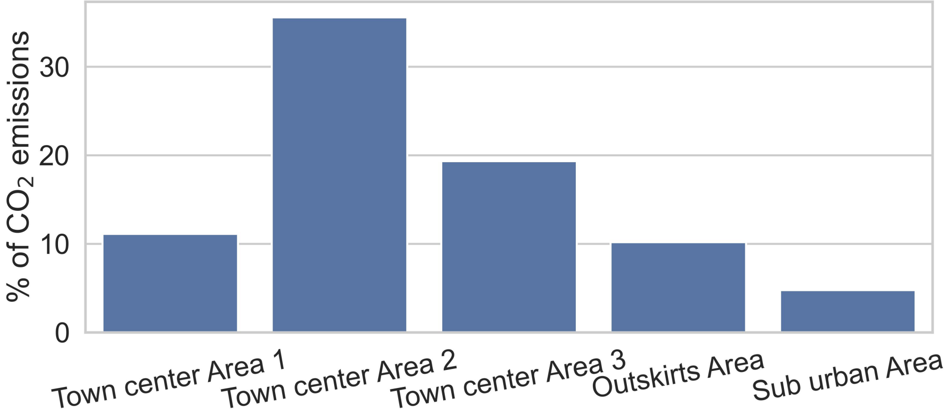


Fig. 6.9 Percentage reduction of CO_2 emissions enabled by a shared PV system over a year for each area of interest.

While the proposed framework is agnostic to the precision of the input data, it is worth noticing how the precision of the result is strictly related to the precision of the input data provided by the user. Regarding the proposed test, as far as census data are concerned, population censuses are the most important statistical data used by policy-makers at all levels of government, as well as private businesses, households, nonprofit organizations, and researchers. It is well known that census data include several sources of problems, such as coverage (omissions or duplications) errors [157]. Moreover, even if the demographic study often validates census data to gauge their quality, this analysis is out-of-scope for our application. Consequently, we believe that census data are an essential source of information for our study. Regarding the DSM data, we decided to use data with a resolution of 0.5 meters. A higher resolution would allow us to obtain a more precise evaluation; however, a higher precision would also imply a higher computational cost. In addition to that, high-resolution data are rare, often expensive, or difficult to obtain. On the other hand, a lower resolution reduces the precision of our evaluation since the identification of the suitable areas, and the estimation of the shadows would be less precise. Thus, for our case studies, the resolution of 0.5 meters represented an excellent tradeoff to obtain sufficient precision. Moreover, our framework can manipulate DSM data with any resolution; consequently, users can choose the precision level they want to achieve and provide the DSM data accordingly.

6.6 Conclusion

This chapter proposes a framework that combines geographical, meteorological, and demographic information to identify optimal photovoltaic installations, evaluate the subsequent benefits for the community members in reducing the energy withdrawn from the power grid, and verify the potential benefits for the environment. To optimize the positioning process of our PV panels, we use GIS technologies and historical data, and we enable the possibility of connecting PV modules located on contiguous buildings. We use real disaggregated consumption data to evaluate the expected power demand and power saving, and estimate the reduction in terms of emission of pollutants. We verify our conjectures on different real scenarios in and around Turin, a large city in the northwest region of Italy. The promising results lead the way to real applications in the field.

Chapter 7

Distributed software platform for raster analysis

7.1 Introduction

During the work made for the case studies of the previous chapters 4,5,6, we had to face a major challenge: the management and processing of the huge amount of raster data involved in the calculation. For all the previous works we had to use a workaround that consist in cropping the entire raster image of the area under study to extract only the areas of the roof towards the south. While this is a reasonable choice since those areas are the most productive ones this reduces the possibilities of further analysis on other areas of the roof. However, raster images are quite common in many fields such as cartography, geography, remote sensing, and environmental science. We noticed however that while the literature offers various solutions, none of them was suitable for our particular purposes, in general the proposed solutions suffer two main issues: i) the setup of the system is not simple for researchers that are not skilled in ICT system mamagement or ii) a solution is tailored to a particular use case and it is not trivial to extend. Therefore the objective of this work was to propose a framework for the management and processing of geographical raster data, which can easily be deployed and flexible enough to serve multiple research objectives and fields. This chapter will proceed as follows: Section 7.2 will present the main solution available nowadays, Section 7.3 will present the format of the data involved and the technologies used in the proposed framework, Section 7.4 is devoted to the explanation of the procedures necessary to perform any data processing. Finally Section 7.5 and 7.6 will present the result for the test case proposed and discuss the possible upgrade that could be done on the proposed solution.

7.2 Literature review

Relational Database Management System

A relational database management system (RDBMS) is based on the relational model, which is not suitable for multi-dimensional data, such as scientific raster data or spatial and temporal ordering data. However, researchers have addressed this limitation by developing solutions to store spatial data within RDBMSs. For example, PostGIS is a free spatial database extension for PostgreSQL that allows users to create location-aware queries using SQL and build their own mapping backends for raster analyses, routing applications, and more. In one study, Davies et al. implemented the storage of MODIS fire archive using the PostGIS extension. Another application of RDBMSs for spatial data is the use of MySQL as a backend for managing large raster data in WebGIS. Scalability is an important factor for handling big data, and while RDBMSs can scale up with expensive hardware, they do not perform well with commodity hardware in parallel. To overcome this issue, the Apache Hadoop project provides open-source software that enables distributed processing of large datasets on clusters of commodity computers using simple programming models. Two popular data containers in the Apache Hadoop ecosystem, Hive and Spark, have been used for processing geospatial data. For instance, SciHive extends Hive to implement a scalable, array-based query system for parallel processing of raw array datasets. Hadoop-GIS integrates indexing techniques into Hive to improve the efficiency of spatial data queries. SciSpark utilizes Spark to create a scalable scientific processing platform for interactive computation and exploration of large raster data. To enhance query performance, Hu et al.[158] developed a multi-dimensional index to enable Spark to natively support array-based datasets stored in HDFS. While frameworks like SciHive and SciSpark achieve scalability, they are not as mature as traditional RDBMSs in terms of system stability, usability, and maintenance, mainly due to issues such as Spark's memory usage impacting latency and cluster performance.

Array-Based DBMS

Traditional RDBMSs are not efficient in handling ordered raster data like satellite images and climate simulation data. As a result, Array DBMSs have gained popularity for managing big scientific data. One pioneering system in this area is PICDMS, which processes image datasets using a grid-based database structure. Another notable Array DBMS is Rasdaman, which has a long history and offers a mature implementation with optimized storage layout, query language, performance, and evaluation. With the advancement of GIScience research, more libraries and software have emerged to support array-based datasets. Another open-source system, SciDB, primarily caters to scientific domains dealing with very large-scale array data, such as astronomy, remote sensing, climate modeling, and bio-science information. It supports a nested-array data model, science-specific operations, uncertainty, lineage, named versions, and user-defined functions, which allow users to implement their own array functions for specific projects.

NoSQL DBMS

NoSQL databases are designed to provide higher performance, elastic scalability, and flexible data modeling. These databases fall into categories such as key-value, document, column, and graph, and they excel at storing large volumes of semi-structured or unstructured data with high Read/Write throughput. Key-value databases like Cassandra offer improved performance for specific use cases such as genomic data management compared to relational databases. However, they have limited functionality beyond key-value storage and lack support for relationships. Other key-value systems include Amazon DynamoDB, and CouchDB. Document-based databases like MongoDB are more effective for managing geospatial data due to their flexible query capabilities and native support for geospatial data formats like GeoJSON. MongoDB has been used to store and access climate satellite data, resulting in significantly improved performance compared to SQL databases. However, document databases do not support relationships and joins like relational databases, and they also lack native support for multidimensional data. Column-based NoSQL databases, such as HBase, offer good scalability and efficient performance for specific query types like range and k-nearest neighbor queries but require a mapping layer to handle highly connected geospatial data. Examples of column-based systems include Google BigTable, Cloudata, and Cassandra (which is also a key-value database).

7.3 Enabling technologies

The framework proposed in this chapter is composed of different state-of-the-art technologies that enable easy customization, deployment, and maintenance. In this section, we will briefly discuss the type of data that must be handled and which technical solutions have been adopted to allow the deployment of the system.

7.3.1 Data format

For the analysis of the energy production of the PV the framework has to deal with two main types of data format: i) *raster data* and ii) vector data. The raster data are geo-referenced images, which means that are pixel-based data with a finite resolution and they also include information on the geographical coordinates of the area that is represented. These images can contain different kinds of information such as altimetry, solar irradiance, hydrology values etc. The resolution of these images is usually quite high and thus produces large-size files that can be difficult to manage. In the proposed framework raster data are used to store information of the DSM and the irradiance values of the area under analysis. Vector data are instead geometrical data, which means that they have a virtually infinite resolution and are used to represent geometrical objects such as polygons, lines, points, and so on. As shown in Figure 7.1 the same type of data can be represented both as a vector or as a raster, according to the needs and one format could be preferred and is some cases there is the need to convert data to one format to the other. In the proposed framework vector data are used to store information regarding the surfaces available for the PV installation or the footprint of the PV modules under test.

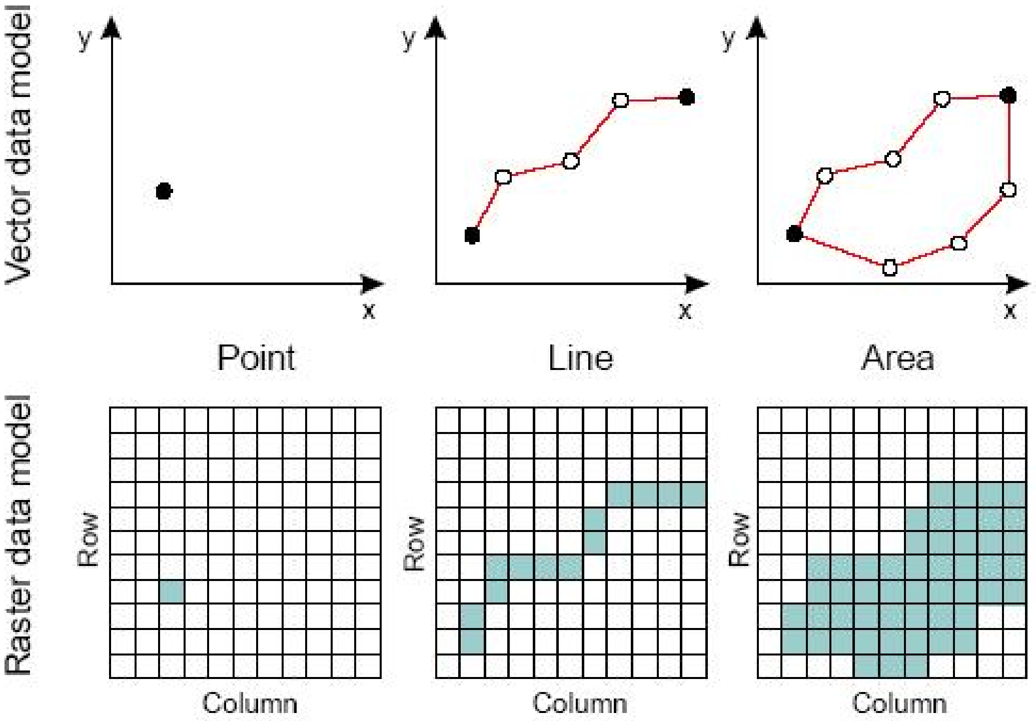


Fig. 7.1 Different handling of a geometric object using vector and raster formats

7.3.2 Distributed architecture

Docker

Docker is an open-source platform that simplifies the deployment and management of applications. It achieves this through containerization, which involves packaging an application and its dependencies into a standardized unit called a container.

Containers are self-contained and portable entities that include all the necessary components to run an application, such as the code, runtime, system tools, and libraries. Docker provides tools and features to create, distribute, and manage these containers effectively.

Key components of Docker include Docker Engine, which manages containers on a host system, Docker Images that serve as templates for containers, Dockerfile for defining container build steps, Docker Registry for storing and sharing images, and Docker Compose for managing multi-container applications.

Using Docker offers benefits like faster application deployment, efficient resource utilization, scalability, and consistency across different environments. It simplifies software delivery and promotes a more streamlined development workflow.

Overall, Docker has become a popular choice for developers, researchers and system administrators as it streamlines the deployment and management of applications in a consistent and reproducible manner.

Apache Hadoop

Apache Hadoop is an open-source framework designed for distributed storage and processing of large-scale datasets across clusters of computers. It provides a reliable, scalable, and fault-tolerant solution for handling big data.

The core components of Apache Hadoop are:

- Hadoop Distributed File System (HDFS): It is a distributed file system that stores data across multiple machines in a cluster. HDFS provides high-throughput access to data and ensures data reliability by replicating blocks across multiple nodes.
- Yet Another Resource Negotiator (YARN): YARN is the cluster management layer of Hadoop. It is responsible for resource allocation and scheduling tasks across the cluster. YARN allows different data processing frameworks, such as MapReduce, to run simultaneously on the same cluster.
- MapReduce: It is a programming model and processing engine for distributed data processing in Hadoop. MapReduce divides large datasets into smaller chunks, processes them in parallel across the cluster, and then combines the results. It simplifies the development of distributed data processing applications.

The functionality of Apache Hadoop revolves around its ability to process vast amounts of data efficiently and in a fault-tolerant manner. Some key aspects of Hadoop's functionality include:

- Scalability: Hadoop can scale horizontally by adding more machines to the cluster, allowing it to handle and process large volumes of data.

- Fault Tolerance: Hadoop achieves fault tolerance by replicating data across multiple nodes in the cluster. If a node fails, the data can be retrieved from other replicas.
- Data Locality: Hadoop optimizes data processing by moving the computation closer to the data. It schedules tasks to run on nodes that already have the required data, minimizing data transfer over the network.
- Data Processing: Hadoop's MapReduce model enables parallel processing of large datasets, making it suitable for tasks like batch processing, log analysis, data aggregation, and more. It provides a programming abstraction that handles the complexities of distributed computing.
- Ecosystem: Hadoop has a rich ecosystem of tools and frameworks built around it, such as Apache Hive for SQL-like queries, Apache Pig for data flow scripting, Apache Spark for in-memory processing, and Apache HBase for NoSQL database operations. These tools extend Hadoop's capabilities and make it more versatile for different data processing needs.

In summary, Apache Hadoop is a powerful framework for distributed storage and processing of big data. It offers scalability, fault tolerance, and a rich ecosystem of tools, making it suitable for various data-intensive applications.

Apache Spark

Apache Spark is an open-source distributed computing system designed for processing and analyzing large-scale datasets. It provides a fast and flexible framework for big data processing and is known for its speed, ease of use, and versatility.

Some key features and concepts of Apache Spark include:

- Resilient Distributed Datasets (RDD): RDD is the fundamental data structure in Spark. It represents an immutable distributed collection of objects that can be processed in parallel across a cluster. RDDs provide fault tolerance and can be cached in memory for faster processing.
- In-Memory Processing: Spark leverages in-memory computing to store intermediate data in memory, which significantly speeds up data processing compared to traditional disk-based systems.

- Data Processing APIs: Spark provides high-level APIs in different programming languages, including Scala, Java, Python, and R. These APIs allow developers to write data processing and analysis tasks using a rich set of built-in functions and transformations.

- Spark SQL: Spark SQL is a module in Spark that enables querying structured and semi-structured data using SQL syntax. It allows developers to perform SQL-like operations on RDDs and seamlessly integrate SQL queries with Spark applications.

- Spark Streaming: Spark Streaming enables real-time processing and analysis of streaming data. It allows developers to ingest data from various sources, such as Kafka or Flume, process it in mini-batches, and perform near-real-time computations.

- Machine Learning Library (MLlib): MLlib is a scalable machine learning library built on top of Spark. It provides a wide range of machine learning algorithms and utilities, making it easier to develop and deploy machine learning models at scale.

- Graph Processing (GraphX): GraphX is a graph processing library in Spark. It allows users to create and manipulate graphs efficiently, perform graph algorithms, and integrate graph processing with other Spark operations.

Apache Spark's versatility and performance make it suitable for a wide range of use cases, including batch processing, interactive queries, real-time analytics, machine learning, and graph processing. It can be deployed on various cluster management systems like Apache Hadoop YARN, Apache Mesos, or in standalone mode.

To summarize, Apache Spark is a powerful and flexible distributed computing system that enables efficient processing and analysis of big data. It provides a unified framework for different data processing tasks, offers in-memory computing, supports various programming languages, and has specialized libraries for machine learning and graph processing.

RasterFrames

Together with the software dedicated to the creation and management of the distributed platform other software are necessary to allow the parallel processing of rater data in a distributed environment. Among the possible framework that offers this feature we selected RasterFrames. Rasterframes can be deployed in a distributed architecture, its based on data frames and allows the analysis and the query of raster data within Apache Spark. It is capable of ingestion of various raster file formats (GeoTiff in particular) and supports GeoJSON. It offers a conspicuous number of functions that can be used to filter transform, resample, and analyze raster images. Among the various possibilities available RasterFrames was preferred due to its built-in capabilities of analyzing raster data organized on a temporal scale.

7.3.3 Additional tools (Swarmpit and Jupyter Notebook)

Swarmpit offers a user-friendly and straightforward interface for effectively managing your Docker Swarm cluster. It provides convenient control over various aspects such as stacks, services, volumes, networks, and more. By connecting your Docker Registry, you can effortlessly deploy private repositories onto the Swarm. Additionally, you have the advantage of securely sharing this management console with your entire team.

Jupyter Notebook is an interactive computing environment that allows users to create and share documents containing live code, visualizations, explanatory text, and more. It is a popular tool among data scientists, researchers, and programmers for data exploration, prototyping, and collaborative work. In a Jupyter Notebook, code is organized into cells, which can be executed individually or as a whole. This interactive nature enables users to experiment with code, modify it, and immediately see the results. The output of code cells, including visualizations and data tables, is displayed inline within the notebook, making it easy to understand and analyze the data. Beyond supporting multiple programming languages like Python, R, and Julia, Jupyter Notebook also allows the inclusion of formatted text, equations, images, and interactive widgets. This makes it a versatile platform for creating comprehensive and interactive documents that combine code, analysis, and explanations. Moreover, Jupyter Notebooks can be shared with others, facilitating collaboration and reproducibility. Notebooks can be exported to various formats such as HTML, PDF, and Markdown, making it straightforward to present or publish your work. Overall, Jupyter Notebook provides a flexible and interactive environment for data exploration, analysis, and sharing, making it a valuable tool in the field of data science and beyond.

7.4 Methodology for the creation of raster processing pipelines

The objective of the application is to calculate the GHI over the roof of the particular area under analysis. The operations to execute are quite trivial and can be intended as matrix multiplication and additions. However, this operation needs to be executed on many files with a considerable dimension. In addition, during this operation, we need to manipulate raster and vector data which in principle are not directly compatible. Rasterframes solve the latter challenge by converting the vector data into raster data. This section presents the different components of the processing pipeline used to assess the capabilities of the proposed platform. Despite the specific use case involved in the testing, the processing executed on the data is composed of operations that are extremely common when working with raster and vector data. This flexibility is one of the advantages of this platform over the previous solution which was tightly coupled with the use case context.

7.4.1 Processing pipeline

This subsection analyzes the different steps that compose the testing pipeline that have been used to test the capabilities of the platform

Raster pre-processing

The first operation that needs to be executed is the compression of the raster data in order to reduce memory usage. Therefore the original data were converted from the GeoTiff format into the Cloud Optimized GeoTiff (COG), which is a common format for GIS framework and is optimized for the cloud environment. The compressed data are then uploaded on the HDFS which takes care of ensuring the needed redundancy to prevent data losses in case of a nod failure. After this preliminary step RasterFrames demands the creation of a file in the CSV format which contains the information that will be needed to retrieve this file during the calculations. For the test case this file, named *catalog*, contains a line of for each couple of rasters (one with the data for the beam irradiance and one with the data for the diffused irradiance). This line contains:

- the date in the format YYYY-MM-DD HH:MM:SS
- the day of the year (from 1 to 365)
- the month
- the hours
- the minutes
- the name of the beam irradiance file
- the name of the diffused irradiance file

An example of the content of such catalof file is shown in the Table 7.1.

date	**year_day**	**month**	**hour**	**minute**	**beam_name**	**diff_name**
2013-01-17 04:00:00	17	1	4	0	beam_17_0400.tiff	diff_17_0400.tiff
2013-01-17 04:15:00	17	1	4	15	beam_17_0415.tiff	diff_17_0415.tiff
2013-01-17 04:30:00	17	1	4	30	beam_17_0430.tiff	diff_17_0430.tiff
2013-01-17 04:45:00	17	1	4	45	beam_17_0445.tiff	diff_17_0445.tiff
2013-01-17 05:00:00	17	1	5	0	beam_17_0500.tiff	diff_17_0500.tiff

Table 7.1 The table contains the first lines of the *catalog* file used for the tests

Connection to Apache Spark

Once the files needed by Rasterframes were ready, we defined the parameters for the connection to Apache Spark. The main parameters that need to be specified are the version and the location of RasterFrames and pyRasterFrames to be used during the execution, Spark allows also for precise and detailed customization of various other parameters. However, after some preliminary tests, we noticed that for such a small deployment the benefits were not improving enough the performance to justify the time spent to carefully optimize the execution. In addition to that, maintaining a default configuration allows greater flexibility for the platform to be used for other case studies.

Definition of the input data

Once the data are available and the Spark execution has been configured we can proceed with the definition of the data that needs to be used in the calculation. For our case study, in particular, we need to specify the *catalog* file generated in the first step and the vector data that needs to be used to filter the raster data and evaluate only the portion of interest. After that our use case also needs data regarding the altitude of the average altitude of the area of interest. This information is obtained by evaluating the altitude of the centroid of that area and by performing a request to a public API [159] to obtain the altitude of this point.

GHI calculation and statistics

The following step consisted of the implementation of the model proposed in [98, 3] using pySpark to allow the parallel calculations. However, in addition to the basic evaluation of the evolution in time of the GHI for the area of interest, we also tested the calculation of its average value and the 75-th percentile. These additional tests were added to show the advantages of using this new framework over the previous solution which was only capable of calculating the evolution of GHI over time.

7.5 Experimental results

7.5.1 Testing setup

The platform described in Section 7.3 and Section 7.4 has been deployed to OpenStack on 9 virtual machines: 1 Master node and 8 worker nodes all running Ubuntu 20.04. The specifications of the two types of machines were the following

1. Master node
    - vCPU: 16 Intel Xeon 2.4Ghz
    - RAM: 64 Gb
    - Storage 256 GB
2. Worker node
    - vCPU: 4 Intel Xeon 2.4Ghz
    - RAM: 8 Gb
    - Storage 256 GB

Using the *compose* plugin for Docker we were able to select on which node the different applications must be deployed. In particular, the whole infrastructure was configured as follows:

- Master Node
    - Docker Registry
    - Apache Spark Master
    - Hadoop HDFS namenode
    - Swarmpit Dashboard
    - Jupyter Notebook
- Worker node
    - Swarmpit agent
    - Apache Spark worker
    - Hadoop HDFS datanode

To evaluate the performance of the platform, we run the standalone version of the PV simulator [77] on an individual server running Ubuntu 20.04 with an Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz and 16 GB of RAM.

7.5.2 Results

Using such deployment 3 testing data process pipelines were applied while changing the number of worker nodes in use from 3 to 5 to assess the performance improvement obtained by increasing such number. The rasters used to perform this calculation are the same used for the work described in Chapter 5 while the vector data used refers to the configuration in the same work, in particular the one with the highest number of panels placed. For each configuration, we calculated the evolution of the GHI values during one year, the average GHI values for 1 year and the 75th percentile of the GHI over 1 year. While the output of this calculation is strongly related to the PV use case the procedures themselves are quite general and were used to test the capabilities of the platform for the calculation of the evolution of a raster value over time and its statistic. Moreover, the calculation of the percentile is not a native function of RasterFrames therefore it also allowed us to assess the extendability of the tool. Table 7.2 shows the execution time of the calculation of the yearly evolution of the GHI during one year. Such table contains the statistics of the test executed with the same processing pipeline but on two different types of input data: the full raster of the area of interest and the "cropped" version that contains only the areas of the roof that are actually useful for the final output. It is worth noticing that the execution time of the "cropped" tests also includes the execution time of the pre-processing required to extract the areas suitable for the PV deployment from the full rasters. The Table 7.2 shows that when the number of nodes increases we do not have any advantages in using the "cropped" version of the rasters. This behavior can be related to the fact that when the number of nodes increases each node can concentrate on a smaller portion of the rasters thus the benefits related to having rasters of smaller size keeps decreasing. We can therefore notice that while the crop of the raster images was necessary with the previous approach in which the calculation was executed on an individual server, with this proposed platform this pre-processing actually does not bring any benefit. Moreover, by keeping the entire rasters, we increase the flexibility of the platform which can be used for other analyses outside the PV topic.

After this preliminary analysis, we tested the same procedure while increasing the number of worker nodes dedicated to the calculation to understand how this number influences the execution time. The result of these tests is shown in Figure 7.2 which shows how, as expected, the execution time decreases when the number of worker

# of nodes	Type of raster	execution time (s)	CPU usage (%)	RAM usage (%)
3	cropped	1310.38	82.2812	39.5415
	full	1304.92	83.6644	39.5602
4	cropped	1345.41	81.3759	39.6968
	full	1025.06	75.2154	52.4585
5	cropped	1000.1	67.7057	48.2
	full	819.878	74.6744	47.3105

Table 7.2 Execution time and statistics of the execution of the GHI calculation on the proposed platform

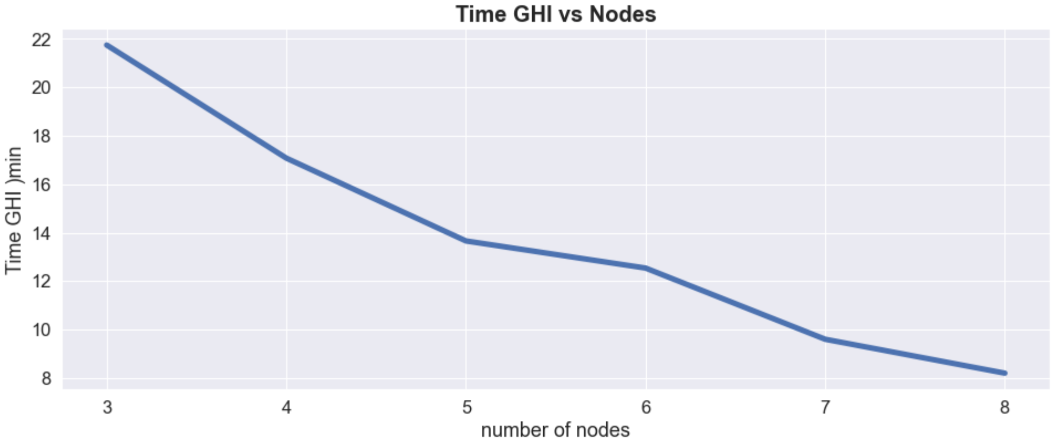


Fig. 7.2 Execution time for the calculation of the GHI traces for one year

nodes increases. However, it is interesting to notice how the relative improvement of the execution time keeps decreasing when increasing the node. This, therefore, suggests that the number of nodes to be used highly depends on the objective of the application. The second test consisted on the calculation of the average value of the ghi for an entire year. This procedure was not achievable on a setup with an individual server since the calculation requires a huge amount of memory and therefore all the trials were automatically stopped for memory issues. Figure 7.3 shows how this calculation was achievable only when the number of nodes was greater than 5.

Finally, the calculation of the 75-th percentile for the roof of the area of interest was achievable only with 8 worker nodes and took 5 hours and 48 minutes. While the calculation of the 75th-percentile is not a CPU-intensive calculation, it actually

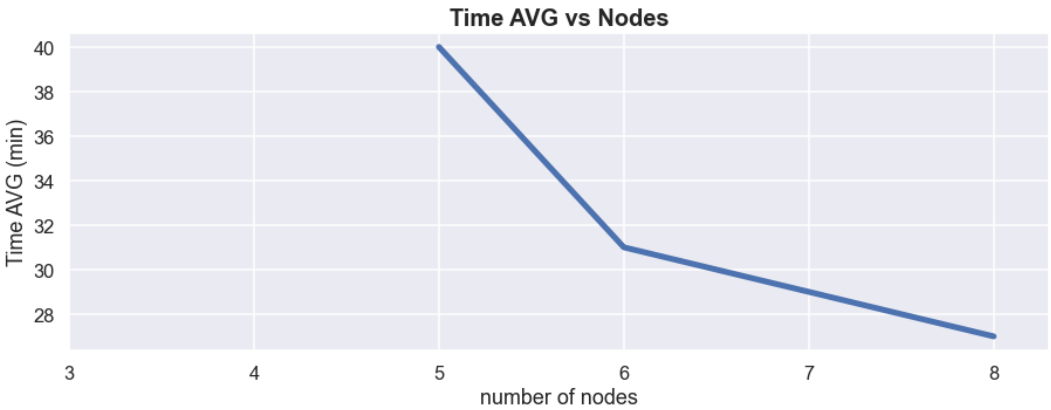


Fig. 7.3 Execution time of the calculation of the average GHI values over one year

requires a high amount of RAM that can be satisfied only by instantiating more worker nodes or by increasing the memory installed on the worker already available. In cloud environments, like the one used in these tests, the best approach depends on the type of infrastructure that is used:

- in a "pays as you go" platform where the price depends only on the usage time, the first solution is the best one since it can speed up the execution thus reducing the cost
- in an Infrastructure as a Service (IaaS) platform, like the one used in this work, where the cost depends on the characteristics of the machine used, the second approach is the best one. Indeed in this case we reduce the cost during the idle time where the machines remanis unused

The summary of execution times for the tested pipelines is summarized in Table 7.3 which shows the performance of the tested pipeline when changing the number of Spark Worker Node available in the platform. This Table suggests how the optimal amount of nodes required by a particular application depends on the type of calculation involved and also on the timing requirements of such an application. While in some cases the results need to be obtained as fast as possible in other situations there may not be the same necessity and therefore increasing the number of nodes may be not necessary

# nodes	GHI execution time	Average GHI execution time	75-th percentile execution time
3	21.75	-	-
4	17.08	-	-
5	16.66	40	-
6	12.54	31.1	-
7	9.6	29.0	-
8	8.20	27.08	348.0

Table 7.3 Summary of the execution times for the tested processing pipelines on the proposed architecture

7.6 Conclusion and future works

The proposed infrastructure shows better performance execution time with respect to the previous approach where the calculation was not executed in parallel and was executed instead on an individual server. Moreover, the technologies and the software used for its development allows the infrastructure to be flexible and to be used for a variety of case studies in various research topics, in which the analysis of geographical raster images has a prominent role.

Chapter 8

Hybrid Multi-Model Co-simulation Infrastructure with HIL

8.1 Introduction

The integration of energy systems will play a fundamental role in the energy transition towards a low-carbon future. The Energy Systems Integration (ESI) is the process of coordinating the operation and planning of energy systems across different energy vectors, infrastructures and sectors that are integrated to optimise the overall resulting complex system [160]. Therefore, ESI will foster the integration of renewable energy sources, distributed multi-generation, energy conversion systems, and energy storage technologies through sector coupling that will reduce the fossil fuel dependencies of the world energy mix, reducing carbon emissions and the energy footprint of present energy systems. This vision will completely change the structure of the energy systems from vertical and centralized energy systems to horizontal, interconnected, and distributed energy systems facing the ESI approach with composite and advanced control strategies, as highlighted by the European Green Deal [161]. Modelling and simulation-based assessments to develop, test and evaluate ESI could provide valuable information to support the system concept exploration and requirements evaluation, reduce the time and cost of prototype development and deployment of innovative technologies, provide data for prediction and/or validation, and better support the planning and operation of these complex energy systems [162].

In the last decades, the research community concentrated its effort on developing standalone simulation tools to cope with the needs of analyzing these complex energy systems [163]. However, standalone solutions have different drawbacks: *i)* existing tools could be hard to integrate with each other due to the usage of widely different technologies, *ii)* proprietary software could not be modified as desired, and *iii)* it is extremely hard to integrate physical devices in the simulation environment [164]. Therefore, standalone simulation fails in describing the multi-disciplinary, multi-domain, and multi-model vision of ESI with the required spatio-temporal scalability. On the other hand, literature has also proposed various solutions to integrate different simulators and to reduce the need for custom solutions which may be too specific for a particular case study, such as[165–167]. However, those tools do not provide real-time simulation for the electrical grid, and they do not support the integration of external simulators or HIL. Instead, other tools such as [168–170] are specifically devoted to the coupling of multi-domain simulators. These tools enable the integration of domain specific models generated with the most diffused modelling tools such as Modelica and EnergyPlus. These tools alone, however, do not completely support a real-time simulator which can be included, but the synchronisation is not ensured in that case.

In literature, co-simulation techniques have been proposed to interconnect different Domain-Specific Programming Language (DSPL) General-Purpose Programming Language (GPPL), and hardware simulators, exploiting each of the respective tool peculiarities in a shared simulation environment capable of analysing concurrent aspects of a Multi-Energy System (MES) [171, 172]. However, co-simulation is a challenging task when dealing with heterogeneous simulation tools. The main challenges regard the communication among different simulation entities and their time regulation and synchronization [173, 174]. Many researchers tried to find solutions to overcome these challenges by developing custom tools to integrate different simulators with each other. A well-known solution in the literature is the Functional Mock-up Interface (FMI) [175] that allows the co-simulation of models between different domain-specific modelling and simulation tools [176].

Other researchers tried different solutions to include Digital Real-Time Simulator (DRTS) in their simulation. In [173, 171], the authors tried to analyze the interaction between the electrical grid and the communication networks. These works interface real-time simulators and communication network simulators which have two different behavior (continuous vs discrete simulation). Most of these works, however, take only into consideration the interaction among these two aspects of the ESI strategy and do not include other energy-related aspects which could have further different timing requirements. Other researchers instead proposed co-simulation infrastructure to enable HIL and Power Hardware-In-the-Loop (PHIL) tests. The works presented in [177, 178] are able to integrate real-time simulator with HIL also when using low-cost hardware. However in most of the studies, including the ones presented in Chapter 2, the HIL is used only with the real-time simulator, and there is no interaction between HIL and a software simulator, such as a control system simulator. Or vice-versa, as the work presented in Chapter 4, 5 and 6, only the software simulations are used to testing design approaches or technologies without interaction with physical hardware. This kind of interaction also requires particular attention in order to maintain the overall synchronization of the co-simulation. Moreover, the interconnection among software, GPPL, and hardware simulators raise complex issues for time regulation and synchronization [179].

This chapter presents an innovative platform that allows the interconnection among multi-model simulation software based on DSPL (e.g. MATLAB Simulink, Modelica, EnergyPlus) and GPPL (e.g. Python, C++, Java) with different spatiotemporal scales, as well as the integration of HIL with commercial Digital Real-Time Simulator (DRTS) (e.g. OPAL-RT) in a distributed and shared co-simulation environment among different computers and servers. The two main contributions of this work were related to: i) the integration of the 3SMA to this new use case, assessing its flexibility and modularity capabilities needed to operate in a different scenario w.r.t. the one presented in the previous chapter and ii) the integration of the same PV simulator used in Chapter 3, 5 and 5

The infrastructure exploits a soft real-time approach where the pure software co-simulation environment runs at the wall-clock time, mimicking a real-world scenario to couple pure software co-simulation with the hard real-time constraints of DRTS. The soft real-time approach is not obliged to precisely respect the real-time constraints and must allow the possibility to run slightly in overrun since a normal software co-simulation environment and its behaviour (e.g. a MES building) does not impact on the fast transient of a power grid. The soft and hard real-time environment communication is ensured by VILLASframework [180], a near real-time middleware, to ensure the correct data exchange among simulation models and platform layers. By employing this novel strategy, the *Hybrid Multi-Model Co-simulation Infrastructure* ensures the correct wall-clock time evolution of the co-simulated MES scenario and respects the real-time constraints of the interconnected DRTS that permits to include fast time-stepped simulations of a power grid model into the MES scenario. Moreover, the DRTS capabilities enable HIL and PHIL tests of real-world hardware, creating a powerful test-bed for innovative power grid technologies and components. To cope with these strict real-time constraints, the software-only co-simulation environment has been stressed and accelerated to reach a low time step duration, around one hundred milliseconds, to deploy a realistic MES scenario where a building is capable of offering demand response and ancillary services (e.g. voltage regulation) to the power grid.

The chapter is structured as follows: Section 8.2 introduces the platform. Section 8.3 presents the scenario exploited for testing the proposed infrastructure. Section 8.4 discusses the experimental results obtained by the co-simulation of the proposed scenario. Finally, Section 8.5 reports concluding remarks and future works.

8.2 Methodology for the implementation of the Co-simulation Infrastructure

The proposed platform makes possible to simulate complex systems towards ESI, offering designers a comprehensive tool to easily interconnect heterogeneous models in the infrastructure in a plug-and-play fashion. The infrastructure leverages a multimodel view in which the designer can choose among several interchangeable versions of the same simulated model, choosing from different engines (i.e. GPPL, software simulators, hardware simulators) depending on the required spatio-temporal scalability. This vision ensures the ability to simulate the fast-to-slow temporal evolution of a ESI scenario in a single co-simulation infrastructure, freeing up the potential to scale the scenario under analysis by choosing the right combination of models in a distributed infrastructure.

The infrastructure also provides the flexibility to develop innovative component models following the appropriate interfaces for data exchange, unlocking simulation capabilities for large-scale scenarios. The designer can choose from simulation software based on DSPL (e.g. MATLAB Simulink, Modelica, EnergyPlus), GPPL models (e.g. Python, C++, Java), and commercial DRTS models (e.g. OPAL-RT).

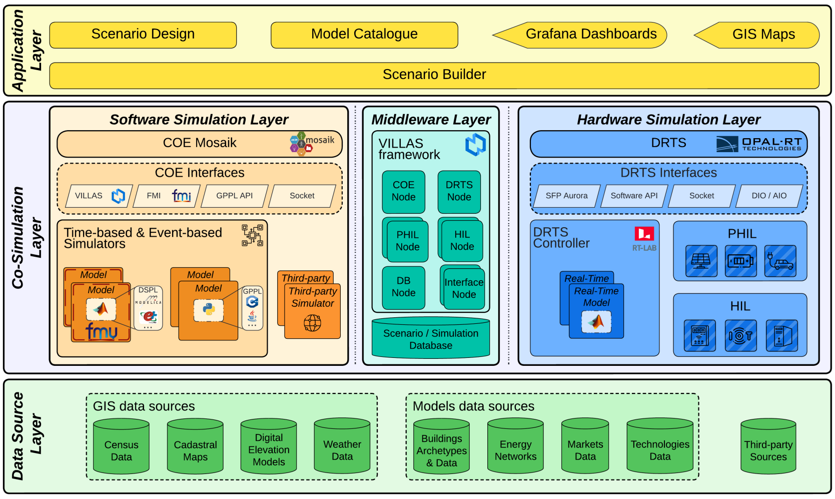


Fig. 8.1 Layered architecture schema of the platform.

Finally, the infrastructure offers the possibility of replacing simulation modules with the equivalent HIL or PHIL device for laboratory testing. In addition, real-world applications and services can be integrated to assess their functionalities in a co-simulation environment.

The infrastructure is presented in Fig. 8.1, consisting of three main vertical layers: *i)* the *Data Source Layer*, *ii)* the *Co-simulation Layer*, and *iii)* the *Application Layer*. The rest of this section will describe each layer in depth.

8.2.1 Data Source Layer

The *Data Source Layer* contains the sources of information needed to describe a scenario, offering standard interfaces to access, query, and retrieve data from other layers of the infrastructure. It includes several Database (DB) modules, which are: *i)* the *GIS data sources* to provide georeferenced data for models of the co-simulation environments that make up a specific scenario (e.g. *Census Data*, *Cadastral Maps*, *Digital Elevation Models*, or *Weather Data*), *ii)* the *Models data sources* to provide basic elements for model definition, such as *Building Archetypes & Data*, *Energy Networks* (e.g., district heating, power network), *Markets Data* and *Technologies Data* of various energy systems, and *iii)* the *Third-party Sources* to support the integration of additional third-party data sources.

8.2.2 Co-simulation Layer

The *Co-simulation Layer* is the core of the platform. It sets up the co-simulation environment retrieving all the required information from the Application Layer (see Section 8.2.3). It is composed of three main horizontal layers, namely *i)* the *Middleware Layer*, *ii)* the *Software Simulation Layer*, and *iii)* the *Hardware Simulation Layer*.

Middleware Layer

It is the central component of the Co-simulation Layer that provides the communication between the soft real-time co-simulation of the Software Simulation Layer and hard real-time simulation environments of the Hardware Simulation Layer exploiting a near real-time approach. It enables this communication by exploiting VILLASframework [181], a toolset for local and geographically distributed real-time co-simulations. It consists of several components called *Nodes* that allow flexible interconnections among different technological software (e.g. GPPL) and hardware (e.g. RTDS, OPAL-RT) components exploiting various communication protocols (e.g. Transmission Control Protocol (TCP), User Datagram Protocol (UDP), MQTT). VILLASframework acts as an adaptation layer, exploiting a near real-time communication that follows the wall clock time, ensuring negligible communication latency among different Nodes. Moreover, VILLASframework allows the interconnection of *PHIL* and *HIL Nodes* such as the smart meter presented in the previous chapter, by exploiting common Internet communication protocols.

Software Simulation Layer

The *Software Simulation Layer* exploits a pure software co-simulation framework based upon Mosaik [170], a Python-based framework originally designed to co-simulate Smart Grid scenarios, which is easily extensible to cope with ESI domains. Its main core is the *Co-simulation Orchestrator Engine (COE) Mosaik* that handles the initialisation, the data exchange management, and the time regulation and synchronisation of simulators and their model instances.

During the initialisation phase, the COE Mosaik passes to all interconnected simulators their parameters (e.g. the number of model instances, time step duration, start date, end date) and model instances input/output relationships with other simulators' model instances. This important task is managed through the Scenario API that offers a common setup procedure for setting the scenario. Moreover, COE Mosaik manages the time regulation and synchronisation of both *Time-based and Event-based Simulators*. Time-based Simulators evolve their model instances with a constant time stepped evolution. Vice versa, Event-based Simulators wait for specific asynchronous events to trigger their model instances' internal state changes and then forward their outputs to other simulators as events.

The data exchange management instead is achieved by exploiting the *COE Interfaces* that allow forwarding the initialization information, the time regulation and synchronisation commands, and the model instances inputs/outputs to each interconnected simulator. Moreover, the COE Interfaces allow the distribution of simulators and their models on different computer clusters, enhancing their vertical and horizontal scalability. Finally, these interfaces could enable the interconnection of *Third-party Simulator* to prevent Intellectual Property Right issues by exploiting the Simulator API. Thus, third-party companies can plug in their own models in a wider scenario without sharing their engines.

Hardware Simulation Layer

It allows the interconnection of commercial Digital Real-Time Simulators (e.g. OPAL-RT or RTDS Technologies) to the proposed co-simulation infrastructure. This layer can run specific models of components that require a hard real-time execution (e.g. power grid, electric vehicle charging system). Moreover, it allows Hardware-In-the-Loop (HIL) and Power Hardware-In-the-Loop (PHIL) to test and validate a real device in a protected virtual environment, avoiding huge deployment costs and the associated risk of deploying the component in a real-world environment.

8.2.3 Application Layer

The *Application Layer* manages the modules describing and composing a complex scenario towards ESI into a co-simulation environment by selecting the proper models and interconnecting them together in a user-friendly and plug-and-play fashion.

The *Model Catalogue* acts as a library that collects all models from already deployed scenarios. Moreover, it gathers information about the configuration setup of each model, such as its time step duration, initial conditions, required inputs, provided outputs, and available data flows with other models.

The *Scenario Design* module instead supports the user definition of a scenario for developing ESI strategies by offering a comprehensive automated tool to interconnect models contained in the Model Catalogue in a plug-and-play fashion. By exploiting each model configuration and setup information that resides in the Model Catalogue, the YAML template guides the platform user in designing the scenario preventing the manual configuration and interconnection of models, which can be error-prone.

Finally, the *Scenario Builder* is in charge of compiling the resulting scenario from the Scenario Design module and communicates its configuration to the Co-simulation Layer, which will instantiate and configure each individual models to concretely execute the co-simulation environment. The Scenario Builder will also validate the proposed scenario by exploiting simulator-specific knowledge (e.g. model configuration and setup information) from the Model Catalogue. The Application Layer also includes *i)* the *Grafana Dashboard* to present co-simulation results by retrieving information from the Scenario/Simulation Database in the Middleware Layer, and *ii)* the *GIS Maps* module instead exploits the georeferenced information from the Data Source Layer and present them into maps to better describe co-simulation results in a spatial scale.

8.3 Scenario and Models Description

This section describes the scenario and simulator models exploited for testing both capabilities and performances of the proposed platform. The scenario proposes the provision of flexibility and effective ancillary services to the power distribution grid via the growing penetration of Distributed Energy Resources (DERs) installed on the building site and the deployment of smart Building Energy Management System (BEMS) and control strategies. In particular, the service is provided directly by the BEMS of a building equipped with a PV and a battery system, coordinated with a Voltage Control System (VCS).

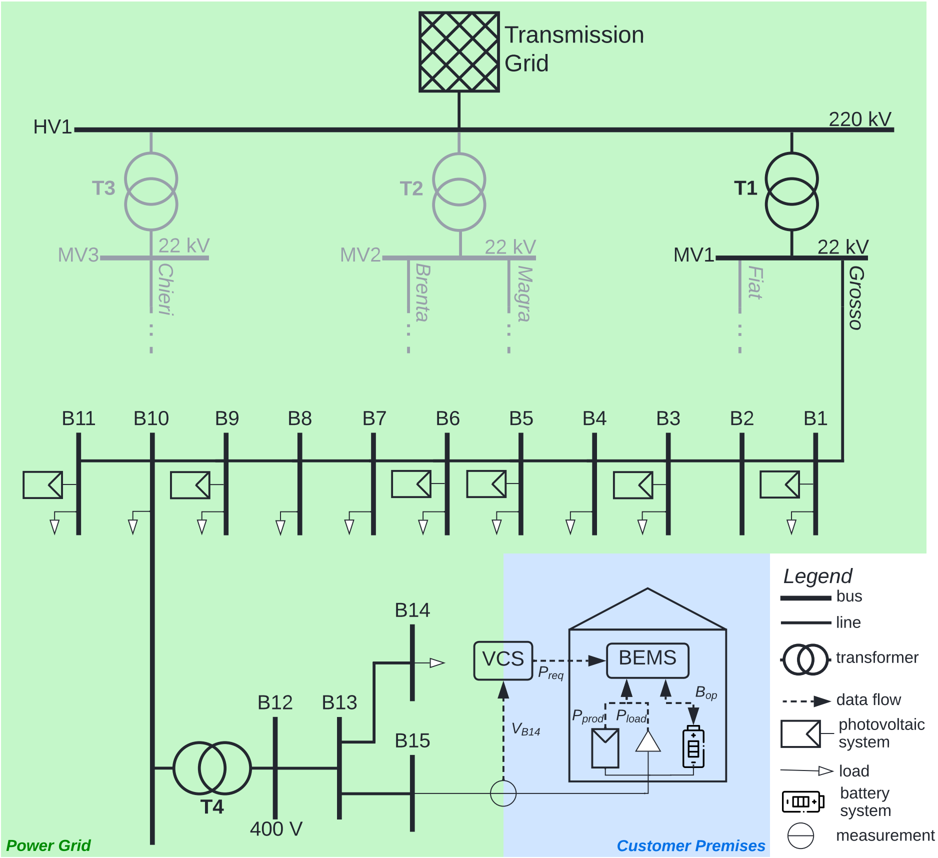


Fig. 8.2 Schema of the MV/LV power grid and the low voltage regulation system coupled with the building energy management system.

Fig. 8.2 shows the overall energy scenario, which consists of an area of interest of a real Italian MV/LV network with real demand and generation profiles obtained from an Italian DSO, and customer premises connected to the distribution grid. The grid was taken from [182] and it consists of five main MV feeders, of which *Grosso* feeder in Fig. 8.2 was considered to perform the scenario simulation. The test feeder consists of 11 buses with 10 equivalent loads and 6 equivalent power injections from PV plants. These injections and withdrawals represent equivalent power system models of the external areas of the grid. The $B10$ bus extends towards the $B13$ bus the LV network where an equivalent LV load is present in bus $B14$ and the building energy system with the LV point of measure and the VCS in bus $B15$. To demonstrate the capability of the VCS and test the platform, the grid was placed under severe energy demand by reasonably increasing the loads causing a decrease of the buses' voltage towards the lower voltage limit. Indeed, the VCS measures

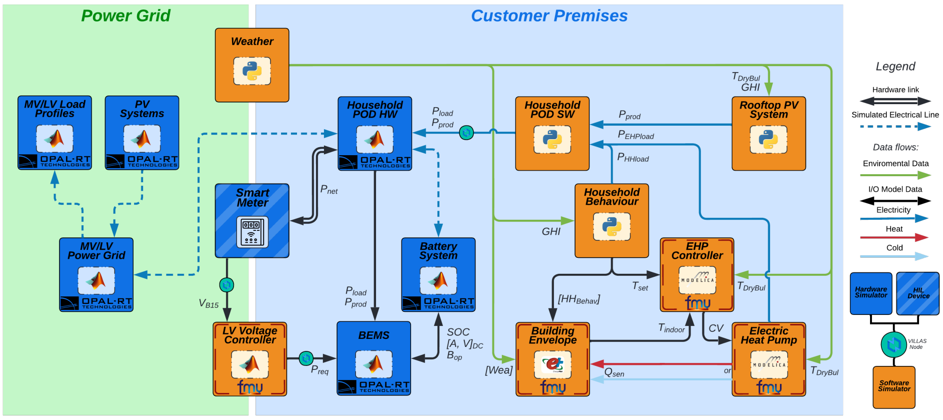


Fig. 8.3 Simulator block diagram of the energy scenario designed within the platform.

the voltage across the LV bus *B*15 and sends the power requests to the BEMS only if the measured voltage exceeds the tolerance of 10% around the nominal voltage value in both direction, i.e. 0.9 *p.u.* and 1.1 *p.u.*, in compliance with the European standard EN 50160 CENELEC [183]. After receiving the power request, the BEMS evaluates the availability of the internal energy resources (i.e., PV production and battery capacity) based on the building load characteristics to decide if the building can provide the total amount of the power requested or a part of it and for how long. The VCS is based on a fuzzy logic Proportional Integral Derivative (PID) controller that can operate at different sample times.

Each element constituting the energy scenario was modelled and implemented into the platform. Fig. 8.3 depicts the scheme of simulation blocks and their connections among them building up the scenario. In particular, the blocks represent the cyber-physical components containing the standalone models related to the customer premises and the power grid. The simulation blocks interact with the shared simulation environment linking them through different kinds of connection based on the simulator typologies, i.e., real-time hardware (blue blocks) and pure software (orange blocks). In addition, the blue striped block represents the HIL component, i.e., the Physical Smart Meter presented in the previous chapter.

The following subsections describe the characteristics of the models implemented for the scenario under analysis, which are divided into *i)* Power Grid and *ii)* Customer Premises.

8.3.1 Power Grid

In this scenario, the environment is modelled by the *Weather* module that provides meteorological data to the system, retrieving the information from third-party data sources. The DRTS simulator has been used to execute the simulation of the power grid running the *MV/LV Power Grid* model. The aggregated *MV/LV* Load Profiles and *PV Systems* generation profiles have been used for all the buses with the only exception of the bus $B15$, which is connected with the high-detailed model of the customer premises. A *Physical Smart Meter* device is connected between the grid and the simulated household by means of an analogue output block in the DRTS model [184]. Therefore, the analogue voltage and current measurements coming from the analogue output board are collected from the DRTS. Moreover, the smart meter communicates the Root Mean Square (RMS) values to the other component of the infrastructure that requires the measurement as input via the HIL Node of VILLASframework. A more detailed description of the above-mentioned simulators and the device can be found in our previous works [185, 182, 184], respectively.

The *LV Voltage Controller* implements the VCS, which is based on fuzzy logic PID controller monitoring the voltage V_{B15} and sending the request capacity P_{req} to the BEMS if the voltage exceeds the tolerance. The controller calculates the error value as the difference between the desired min/max tolerated voltage and the measured value V_{B15} and applies a correction based on proportional, integral and derivative terms on the P_{req} that will be requested to not exceed the tolerance and minimise the error over time. The model is developed in Simulink using the Fuzzy Logic and Control System Toolboxes, and integrated via the FMI interface of the COE Mosaik.

8.3.2 Customer Premises

The customer premises represent the building energy system with its physical components modelled by coupling diverse simulators, described in the following sections. In particular, the *Building Envelope* simulates the building thermal dynamic; the *Household Behaviour* simulates the occupancy and the use of light and appliances in terms of energy consumption derived by the action of each inhabitant in the house; the *Rooftop PV System* provides a high-detailed PV profile than the *PV systems* simulator, as it uses more granular spatiotemporal data and detailed technical information of the building rooftop solar potential; finally, the *Electric Heat Pump* (EHP) and its *EHP Controller* simulate the operations of the heating system to control and maintain the desired indoor temperature. A more detailed description of the above-mentioned simulators can be found in our previous works [186, 185], while the interconnections of these models within the scenario are depicted in Fig. 8.3.

The *Battery System* model was designed in MATLAB Simulink and embedded into the DRTS to couple it with the power distribution network in real-time, in order to describe the fast dynamics correctly. The model simulates the dynamic behaviour of a Li-Ion battery, and it can be fully parameterised using a commercial battery datasheet. The main state variables of the battery (i.e., SOC, A and V in Direct Current (DC)) are sent to BEMS that manages the energy fluxes in the building though the battery signal operations B_{op} based on the requests of the VCS. The battery model is set to simulate the charge and discharge of the battery with a resolution of 10 minutes.

The *Household Point of Delivery (POD) HW* represents the physical point of withdrawal and/or injection of electricity into the distribution network. It was modelled in Simulink as a PQ Load element and embedded into the DRTS. In particular, the Household POD receives the power-related data signals from the pure software simulators of the building through COE Node of VILLASframework and translates them into real-time simulated Alternate Current (AC) voltage and current. Moreover, It sends the data of power withdrawal or injection to the BEMS.

The *Household POD SW* provides the data interface between the hardware Household POD and the pure software simulators of the building. It collects the PV electrical generation (P_{prod}), the aggregated household electrical load (P_{HHload}), the Electric Heat Pump (EHP) electrical consumption ($P_{EHPload}$), and main state variables of the battery (SOC, A and V in DC). It was used as a data collector manager returning the simulation results either in run-time or at the end of the simulation. The smart meter simulator can perform the simulation with whatever time resolution without any limitation.

The *Building Energy Management System* is a ruled-based control algorithm that manages the charge and discharge of the battery under the depth of discharge limit, prioritising self-consumption and, eventually, charging the battery only from the surplus of PV production. However, when the BEMS receives the requests for upward or downward capacity from the VCS, it verifies if there is space for providing the requested flexibility or not by directly controlling the battery parameters for further injection/withdrawal or the PV system only for withdrawal.

8.4 Experimental Results

The scenario presented in Section 8.3 was simulated in order to test the functionalities of the presented platform.

8.4.1 Software, Hardware, and Network Setup

The Co-simulation Layer is deployed among five servers and one OPAL-RT OP5700 (i.e. a DRTS), that are all interconnected between them via a 10 Gbps Ethernet switch minimising the latency in data exchange.

The first server hosts the COE Mosaik, its Interfaces, and all the Middleware Layer entities in Fig. 8.1. The other four servers instead host all the software modules in the energy scenario in Fig. 8.3. All the models communicate with the COE Mosaik by implementing the FMI and GPPL API Interfaces.

The OPAL-RT OP5700 hosts the following modules of Fig. 8.3: *i)* MV/LV Power Grid, *ii)* MV/LV Load Profiles, *iii)* Photovoltaic System, *iv)* Household POD, *v)* Battery System, and *vi)* BEMS. To ensure the time synchronization between the server and the OPAL-RT OP5700, they are all equipped with an Oregano Syn1588 Precision Time Protocol (PTP) synchronization board that implements IEEE1588 PTP standard. Finally, the Physical Smart Meter is connected to the OPAL-RT OP5700 via an AIO DRTS Interface from which the meter retrieves the voltage and current of the monitored LV Bus. Then, the meter computes both RMS and phase and sends these results to the rest of the infrastructure via the HIL Node in the Middleware Layer.

8.4.2 Scenario Setup

The standardized YAML configuration files were filled through the *Scenario Design* with all data and parameters required to set up the co-simulation environment, simulators and their models, as well as connections among them as depicted in Fig. 8.3. The relevant model settings are described in the following.

The models related to the customer premises were parameterized and set as in [185]. In particular, the household consists of four members, and the installed PV system provides a total of $10kW_p$ to the premises. The time steps were set considering the capability of the models' solvers and computational effort, as well as the needs for the real-time hardware coupling: EnergyPlus Building Envelope 10 *min*, Modelica Electric Heat Pump 5 *min*, Rooftop PV System 15 *min*, Household Behaviour 10 *min*, Matlab Simulink LV Voltage Controller 100 ms, and Household POD SW 100 ms. In particular, these last two were set with smaller time steps as they shall be coupled with the hardware simulators, as depicted in Fig. 8.3. The *Weather* module provides data to each simulation engine at the requested time step. All the models implemented in OPAL-RT (i.e., the MV/LV Power Grid, the MV/LV Load Profiles, the PV Systems, the Battery System, the BEMS, and the Household POD HW) run with a time step duration of 50 μs. Moreover, it was assumed that for all the connection points of the grid, the power factor is maintained above 0.95. The Battery System is composed of two battery packs in series with a rated capacity of 60 Ah and a nominal voltage of 200 V. The parameters of the discharge characteristics were derived from the built-in Li-Ion battery Simulink model.

In the end, the *Scenario Builder* module parses the YAML configuration files to retrieve all the required information to perform the scenario simulation and automatically distributes them to the Co-Simulation Layer exploiting the API of COE Mosaik, DRTS OPAL-RT, and middleware VILLASframework.

8.4.3 Scenario Results

The scenario was simulated by exploiting the real-time capability of the platform for two consecutive days during the heating season by considering the use cases without voltage control, called BAU, and with the activated voltage control system, called VRS. The main results of the simulation are depicted in Fig. 8.4 (BAU on the left and VRS on the right). It shows the main characterizing variables for comparing the two use cases and brings out the benefits of the control strategies, as well as the delays and noises that derive from the coupling of pure software and hardware simulators. In particular, Fig. 8.4*(a)* and Fig. 8.4*(f)* show the RMS voltage in p.u.

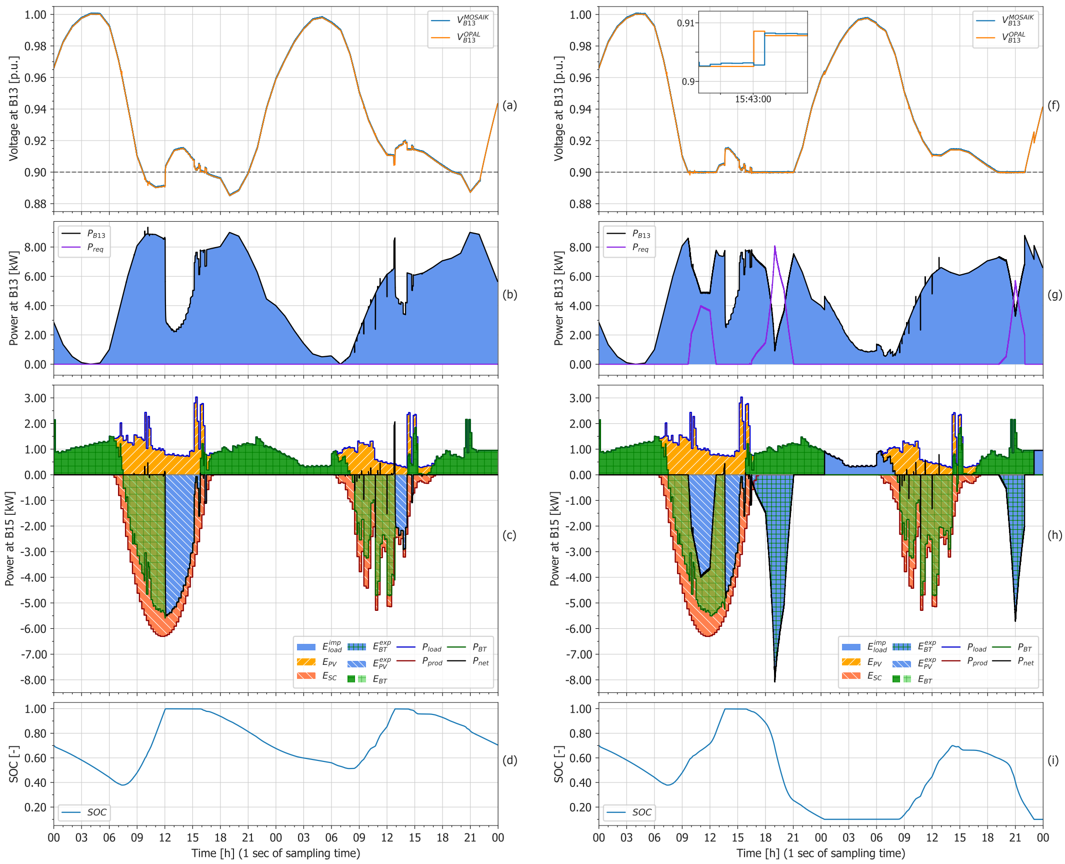


Fig. 8.4 Simulation results of two consecutive days during the thermal season by considering the use cases without voltage control BAU (*on the left*), and with activated voltage control system VRS (*on the right*). In particular, plots (c) and (h) show the power and energy exchanges at bus B15 measured on the customer premises, i.e., related to the battery (P_{BT}, E_{BT} and E_{BT}^{exp}), PV (P_{prod}, E_{PV} and E_{PV}^{exp}), consumption (P_{load} and E_{load}^{imp}), net power and self-consumption (P_{net} and E_{SC}).

measured at bus $B13$. To highlight the round trip time latency due to the HIL and the co-simulation application, the voltage measure was collected directly from the Real-Time Model, V_{B13}^{OPAL}, and indirectly from the Physical Smart Meter, V_{B13}^{MOSAIK}, both in OPAL-RT. Fig. 8.4*(b)* and Fig. 8.4*(g)* show the power measured at bus $B13$, P_{B13}, and the power request, P_{req}, for voltage control, both collected from OPAL-RT. Fig. 8.4*(c)* and Fig. 8.4*(h)* show the power and related energy exchanges at bus $B15$ measured on the customer premises (see the figure caption for more details), and the battery State Of Charge (SOC) in Fig. 8.4*(d)* and Fig. 8.4*(i)*. In particular, the load, P_{load}, and production, P_{prod}, data were collected from Mosaik, while the data related to the battery and the net power and energy exchanges were collected from OPAL-RT. Overall, the data were collected with a sampling time of 1 s. However, it is possible to request information from each module exposing inputs/outputs and parameters or from the simulation environment directly at any sampling time.

By comparing the voltage curves in Fig. 8.4*(a)* and Fig. 8.4*(f)*, the VCS together with the BEMS succeed on maintaining the voltage above the tolerance limit of 0.9 $p.u.$ during the whole period of simulation, as opposed to BAU use case in which the voltage falls under the limit due to the high network withdrawals, especially during the peak hours as depicted in Fig. 8.4*(b)*. Considering the VRS use case, the Fig. 8.4*(g)* and Fig. 8.4*(h)* clarify how VCS and BEMS act to control the voltage on the distribution grid. The first capacity request P_{req} occurs from 09:30 till 12:30 of the first simulated day, amounting to approximately 4 kW at the peak. It can be seen that the request is immediately fulfilled by the PV production E_{PV}^{exp} of the building by injecting directly into the grid rather than charging the battery, as highlighted by the reduction of the stored energy E_{BT} in the request period. By analysing the second grid request of the day, from 16:00 to 21:00, of about 8 kW at the peak, the capacity is provided directly from the battery that is discharging to cover the household demand E_{BT} as well as the external request E_{BT}^{exp}. It is important to note that, unlike the BAU use case, the PV production and battery capacity are exploited when needed and if possible by the VCS causing probable inconveniences for the household. For example, as shown in Fig. 8.4*(h)* and Fig. 8.4*(i)* as opposed to Fig. 8.4*(c)* and Fig. 8.4*(d)*, the battery results completely discharged at about 00:00 and it is not able to cover the household demand till 09:00 of the day after (the SOC reaches the discharge limit of 10%), naturally obliging withdrawal from the grid during this period. This is a cost for the family that must be adequately covered and remunerated in order to sustain the network ancillary service. Moreover, this withdrawal, which happens in VRS use case compared to BAU, does not affect the ancillary service because it occurs at night when the demand for electricity in the grid is low.

This use case also demonstrates the flexibility and modularity of the smart meter prototype describe in the previous chapter. Despite the completely different characteristics of the two case studies, thanks to the design of the device, it was extremely easy to include the libraries and the communication adapter needed to operate in the new use case. Moreover, even considering the device limitation, this scenario demonstrates the benefit that such a low-cost device would provide in the Smart Grid scenario.

8.5 Conclusion

This chapter presented the platform for ESI that enables the coupling of multi-model simulation software with hardware simulators and devices in a shared and distributed co-simulation environment. To couple software-only simulators with the hard real-time world, the infrastructure uses a near real-time middleware based on VILLAS Framework for data processing and exchange between simulation models. In this way, the correct temporal evolution of the co-simulated scenario is ensured and the real-time constraints of the networked DRTS are respected, allowing the inclusion of the fast time-stepped simulation of a power grid model.

An energy scenario is proposed to test both capability and performances of the infrastructure to perform multi-domain analysis of scenarios for ESI by interconnecting different software, devices, and digital real-time simulation environments, extending the capabilities of standalone or monolithic simulations. The scenario evaluates a voltage regulation ancillary service for the power grid distribution network provided directly by the BEMS of a building equipped with a PV and a battery system coordinated with a VCS. The scenario mainly consists of software-only simulators for all the entities and systems composing the customer premises, except for the battery and BEMS simulated into DRTS together with the power grid, and the Physical Smart Meter device in HIL. The scenario results showed the capability of the infrastructure to deal with coupled software and hardware simulators, and HIL devices with good performances, high details, and low co-simulation latencies with an average value of 287.5 ms.

The infrastructure was developed to be used as a virtual test bed for complex ESI by the research community to assess energy systems integration through new enabling technologies, or by system and flexibility operators to design and test new business models, software and hardware prototypes.

References

[1] Julio Romero Agüero. Applying self-healing schemes to modern power distribution systems. In *2012 IEEE Power and Energy Society General Meeting*, pages 1–4. IEEE, 2012.

[2] Mitsubishi Electric. PV-MF165EB3 (165Wp). https://www.mitsubishielectricsolar.com, 2004.

[3] S. Vinco, L. Bottaccioli, E. Patti, A. Acquaviva, E. Macii, and M. Poncino. GIS-based optimal photovoltaic panel floorplanning for residential installations. In *Proc. of DATE*, pages 437–442, 2018.

[4] A Naumann, I Bielchev, N Voropai, and Z Styczynski. Smart grid automation using iec 61850 and cim standards. *Control Engineering Practice*, 25:102–111, 2014.

[5] V Cagri Gungor, Dilan Sahin, Taskin Kocak, Salih Ergut, Concettina Buccella, Carlo Cecati, and Gerhard P Hancke. A survey on smart grid potential applications and communication requirements. *IEEE Transactions on industrial informatics*, 9(1):28–42, 2012.

[6] Lilia Tightiz and Hyosik Yang. A comprehensive review on iot protocols' features in smart grid communication. *Energies*, 13(11):2762, 2020.

[7] PRICE CODE. Communication networks and systems in substations–part 5: Communication requirements for functions and device models, 2003.

[8] Muhammad Kamran. *Fundamentals of Smart Grid Systems*. Elsevier, 2022.

[9] Mehmet Hazar Cintuglu, Osama A Mohammed, Kemal Akkaya, and A Selcuk Uluagac. A survey on smart grid cyber-physical system testbeds. *IEEE Communications Surveys & Tutorials*, 19(1):446–464, 2016.

[10] Yi Wang, Qixin Chen, Tao Hong, and Chongqing Kang. Review of Smart Meter Data Analytics: Applications, Methodologies, and Challenges. *IEEE Transactions on Smart Grid*, 10(3):3125–3148, 2019.

[11] Qie Sun, Hailong Li, Zhanyu Ma, Chao Wang, Javier Campillo, Qi Zhang, Fredrik Wallin, and Jun Guo. A comprehensive review of smart energy meters in intelligent energy networks. *IEEE Internet of Things Journal*, 3(4):464–479, 2015.

[12] Hannah Ritchie, Max Roser, and Pablo Rosado. Renewable energy, Oct 2022.

[13] Laura Mehigan, J Paul Deane, BP Ó Gallachóir, and Valentin Bertsch. A review of the role of distributed generation (dg) in future electricity systems. *Energy*, 163:822–836, 2018.

[14] Simon Funcke and Dierk Bauknecht. Typology of centralised and decentralised visions for electricity infrastructure. *Utilities Policy*, 40:67–74, 2016.

[15] Fabrício Peter Vahl, Ricardo Rüther, and Nelson Casarotto Filho. The influence of distributed generation penetration levels on energy markets. *Energy policy*, 62:226–235, 2013.

[16] Mohammad Al-Muhaini and Gerald T Heydt. Evaluating future power distribution system reliability including distributed generation. *IEEE transactions on power delivery*, 28(4):2264–2272, 2013.

[17] Angel A Bayod-Rújula. Future development of the electricity systems with distributed generation. *Energy*, 34(3):377–383, 2009.

[18] Energy communities.

[19] Jens Lowitzsch, Christina E Hoicka, and Felicia J van Tulder. Renewable energy communities under the 2019 european clean energy package–governance model for the energy clusters of the future? *Renewable and Sustainable Energy Reviews*, 122:109489, 2020.

[20] Hirushie Karunathilake, Kasun Hewage, Walter Mérida, and Rehan Sadiq. Renewable energy selection for net-zero energy communities: Life cycle based decision making under uncertainty. *Renewable Energy*, 130:558–573, 2019.

[21] Christina E. Hoicka, Jens Lowitzsch, Marie Claire Brisbois, Ankit Kumar, and Luis Ramirez Camargo. Implementing a just renewable energy transition: Policy advice for transposing the new european rules for renewable energy communities. *Energy Policy*, 156:112435, 2021.

[22] Deyana Spasova and Sibylle Braungardt. Building a common support framework in differing realities—conditions for renewable energy communities in germany and bulgaria. *Energies*, 14(15), 2021.

[23] Dina Stober, Monika Suškevičs, Sebastian Eiter, Stefanie Müller, Stanislav Martinát, and Matthias Buchecker. What is the quality of participatory renewable energy planning in europe? a comparative analysis of innovative practices in 25 projects. *Energy Research & Social Science*, 71:101804, 2021.

[24] Florian Hanke, Rachel Guyet, and Marielle Feenstra. Do renewable energy communities deliver energy justice? exploring insights from 71 european cases. *Energy Research & Social Science*, 80:102244, 2021.

[25] N Carlisle, J Elling, and T Penney. Renewable energy community: Key elements. Technical report, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2008.

[26] Fabio Cremona, Marten Lohstroh, David Broman, Edward A Lee, Michael Masin, and Stavros Tripakis. Hybrid co-simulation: it's about time. *Software & Systems Modeling*, 18(3):1655–1679, 2019.

[27] A. Ghosal and M. Conti. Key management systems for smart grid advanced metering infrastructure: A survey. *IEEE Communications Surveys Tutorials*, 21(3):2831–2848, thirdquarter 2019.

[28] A Ghasempour and J Lou. Advanced metering infrastructure in smart grid: Requirements, challenges, architectures, technologies, and optimizations. In *Smart Grids: Emerging Technologies, Challenges and Future Directions*. Nova Science Publishers, 2017.

[29] A. Ghasempour. Optimized scalable decentralized hybrid advanced metering infrastructure for smart grid. In *2015 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 223–228, Nov 2015.

[30] M. Nardello, M. Rossi, and D. Brunelli. An innovative cost-effective smart meter with embedded non intrusive load monitoring. In *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, pages 1–6, Sep. 2017.

[31] A. Zidan, M. Khairalla, A. M. Abdrabou, T. Khalifa, K. Shaban, A. Abdrabou, R. El Shatshat, and A. M. Gaouda. Fault detection, isolation, and service restoration in distribution systems: State-of-the-art and future trends. *IEEE Transactions on Smart Grid*, 8(5):2170–2185, Sep. 2017.

[32] Fadi Al-Turjman and Mohammad Abujubbeh. Iot-enabled smart grid via sm: An overview. *Future Generation Computer Systems*, 96:579–590, 2019.

[33] Dlms: Device language message specification | dlms.

[34] Synchronous modular meter - vde fnn.

[35] Ye Yan, Rose Qingyang Hu, Sajal K Das, Hamid Sharif, and Yi Qian. An efficient security protocol for advanced metering infrastructure in smart grid. *IEEE Network*, 27(4):64–71, 2013.

[36] Feng Ye, Yi Qian, and Rose Qingyang Hu. A security protocol for advanced metering infrastructure in smart grid. In *2014 IEEE Global Communications Conference*, pages 649–654. IEEE, 2014.

[37] A. Meloni and L. Atzori. A cloud-based and restful internet of things platform to foster smart grid technologies integration and re-usability. In *2016 IEEE International Conference on Communications Workshops (ICC)*, pages 387–392, 2016.

[38] Songlin Chen, Hong Wen, Jinsong Wu, Wenxin Lei, Wenjing Hou, Wenjie Liu, Aidong Xu, and Yixin Jiang. Internet of things based smart grids supported by intelligent edge computing. *IEEE Access*, 7:74089–74102, 2019.

[39] Yu Yan and Wencong Su. A fog computing solution for advanced metering infrastructure. In *2016 IEEE/PES Transmission and Distribution Conference and Exposition (T D)*, pages 1–4, 2016.

[40] Q. Ou, Y. Zhen, X. Li, Y. Zhang, and L. Zeng. Application of internet of things in smart grid power transmission. In *2012 Third FTRA International Conference on Mobile, Ubiquitous, and Intelligent Computing*, pages 96–100, 2012.

[41] J. Lloret, J. Tomas, A. Canovas, and L. Parra. An integrated iot architecture for smart metering. *IEEE Communications Magazine*, 54(12):50–57, 2016.

[42] Michele De Donno, Koen Tange, and Nicola Dragoni. Foundations and evolution of modern computing paradigms: Cloud, iot, edge, and fog. *Ieee Access*, 7:150936–150948, 2019.

[43] Yu Yan and Wencong Su. A fog computing solution for advanced metering infrastructure. *Proceedings of the IEEE Power Engineering Society Transmission and Distribution Conference*, 2016-July, 7 2016.

[44] Syed Aon Ali Naqvi, Nadeem Javaid, Hanan Butt, Muhammad Babar Kamal, Ali Hamza, and Muhammad Kashif. Metaheuristic optimization technique for load balancing in cloud-fog environment integrated with smart grid. *Lecture Notes on Data Engineering and Communications Technologies*, 22:700–711, 2019.

[45] David Bakken, Alex Askerman, Anurag Srivastava, Patrick Panciatici, Maik Seewald, Frank Columbus, and Song Jiang. Towards enhanced power grid management via more dynamic and flexible edge computations. *2017 IEEE Fog World Congress, FWC 2017*, pages 1–8, 5 2018.

[46] Mohamed Saleem Haja Nazmudeen, Au Thien Wan, and Seyed M. Buhari. Improved throughput for power line communication (plc) for smart meters using fog computing based data aggregation approach. *IEEE 2nd International Smart Cities Conference: Improving the Citizens Quality of Life, ISC2 2016 - Proceedings*, 9 2016.

[47] Md Hussain, MM Beg, et al. Fog computing for internet of things (iot)-aided smart grid architectures. *Big Data and cognitive computing*, 3(1):8, 2019.

[48] Miodrag Forcan and Mirjana Maksimović. Cloud-fog-based approach for smart grid monitoring. *Simulation Modelling Practice and Theory*, 101:101988, 2020.

[49] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang. Intelligent edge computing for iot-based energy management in smart cities. *IEEE Network*, 33(2):111–117, 2019.

[50] Yu Yan and Wencong Su. A fog computing solution for advanced metering infrastructure. In *2016 IEEE/PES Transmission and Distribution Conference and Exposition (T&D)*, pages 1–4. IEEE, 2016.

[51] Andrea Angioni, Gianluca Lipari, Marco Pau, Ferdinanda Ponci, and Antonello Monti. A low cost pmu to monitor distribution grids. In *2017 IEEE International Workshop on Applied Measurements for Power Systems (AMPS)*, pages 1–6. IEEE, 2017.

[52] Paolo Romano, Mario Paolone, Thomas Chau, Ben Jeppesen, and Elias Ahmed. A high-performance, low-cost pmu prototype for distribution networks based on fpga. In *2017 IEEE Manchester PowerTech*, pages 1–6. IEEE, 2017.

[53] Pietro Romano, Mario Paolone, Jack Arnold, and Roberto Piacentini. An interpolated-dft synchrophasor estimation algorithm and its implementation in an fpga-based pmu prototype. In *2013 IEEE Power & Energy Society General Meeting*, pages 1–6. IEEE, 2013.

[54] JCDC Gallano, VJD Malvas, JLF Quirona, RCS Soriano, MC Pacis, and FRG Cruz. Design and implementation of phasor measurement unit with iot technology. In *2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, pages 1–6. IEEE, 2020.

[55] T. Sirojan, S. Lu, B. T. Phung, and E. Ambikairajah. Embedded edge computing for real-time smart meter data analytics. In *2019 International Conference on Smart Energy Systems and Technologies (SEST)*, pages 1–5, 2019.

[56] Yung-Yao Chen, Yu-Hsiu Lin, Chia-Ching Kung, Ming-Han Chung, and I-Hsuan Yen. Design and implementation of cloud analytics-assisted smart power meters considering advanced artificial intelligence as edge analytics in demand-side management for smart homes. *Sensors*, 19(9), 2019.

[57] Paolo Attilio Pegoraro, Alessio Meloni, Luigi Atzori, Paolo Castello, and Sara Sulis. Pmu-based distribution system state estimation with adaptive accuracy exploiting local decision metrics and iot paradigm. *IEEE Transactions on Instrumentation and Measurement*, 66(4):704–714, 2017.

[58] Marco Pignati, Miroslav Popovic, Sergio Barreto, Rachid Cherkaoui, German Dario Flores, Jean-Yves Le Boudec, Maaz Mohiuddin, Mario Paolone, Paolo Romano, Styliani Sarri, et al. Real-time state estimation of the epfl-campus medium-voltage grid by using pmus. In *2015 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5. IEEE, 2015.

[59] Roy T Fielding and Richard N Taylor. Principled design of the modern web architecture. *ACM Transactions on Internet Technology (TOIT)*, 2(2):115–150, 2002.

[60] Patrick Th. Eugster, Pascal A. Felber, Rachid Guerraoui, and Anne-Marie Kermarrec. The many faces of publish/subscribe. *ACM Comput. Surv.*, 35(2):114–131, June 2003.

[61] MQTT. http://mqtt.org, Accessed on June 2021.

[62] USB 12-Bit DAQ Devices with 8 Analog Inputs - Measurement Computing, Accessed on June 2021.

[63] Abouzar Estebsari, Enrico Pons, Ettore Bompard, Alireza Bahmanyar, and Sadegh Jamali. An improved fault location method for distribution networks exploiting emerging lv smart meters. In *2016 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS)*, pages 1–6. IEEE, 2016.

[64] Marco Pau, Edoardo Patti, Luca Barbierato, Abouzar Estebsari, Enrico Pons, Ferdinanda Ponci, and Antonello Monti. A cloud-based smart metering infrastructure for distribution grid services and automation. *Sustainable Energy, Grids and Networks*, 15:14–25, 2018.

[65] Marco Pau, Edoardo Patti, Luca Barbierato, Abouzar Estebsari, Enrico Pons, Ferdinanda Ponci, and Antonello Monti. Design and accuracy analysis of multilevel state estimation based on smart metering infrastructure. *IEEE Transactions on Instrumentation and Measurement*, 68(11):4300–4312, 2019.

[66] Luca Barbierato, Abouzar Estebsari, Lorenzo Bottaccioli, Enrico Macii, and Edoardo Patti. A distributed multimodel cosimulation platform to assess general purpose services in smart grids. *IEEE Tran. on Industry Applications*, 56(5):5613–5624, 2020.

[67] Mininet, Accessed on June 2021.

[68] Vivien GUEANT. iperf, Accessed on June 2021.

[69] Eclipse Mosquitto, Accessed on June 2021.

[70] BNC-2120.

[71] DAQCard-60602.

[72] IEEE 802.3-2018 - IEEE Standard for Ethernet, Accessed on June 2021.

[73] Daniele Salvatore Schiera, Luca Barbierato, Andrea Lanzini, Romano Borchiellini, Enrico Pons, Ettore Bompard, Edoardo Patti, Enrico Macii, and Lorenzo Bottaccioli. A distributed multimodel platform to cosimulate multienergy systems in smart buildings. *IEEE Transactions on Industry Applications*, 57(5):4428–4440, 2021.

[74] E. S. Kumar and B. Sarkar. Improved modeling of failure rate of photovoltaic modules due to operational environment. In *Proc. of ICCPCT*, pages 388–393, 2013.

[75] GDAL — GDAL documentation.

[76] Robert Margolis, Pieter Gagnon, Jennifer Melius, Caleb Phillips, and Ryan Elmore. Using GIS-based methods and lidar data to estimate rooftop solar technical potential in US cities. *Environmental Research Letters*, 12, 2017.

[77] Lorenzo Bottaccioli, Edoardo Patti, Enrico Macii, and Andrea Acquaviva. GIS-based software infrastructure to model pv generation in fine-grained spatio-temporal domain. *IEEE Systems Journal*, 2017.

[78] S. Freitas, C. Catita, P. Redweik, and M.C. Brito. Modelling solar potential in the urban environment: State-of-the-art review. *Renew. Sustainable Energy Rev.*, 41:915–931, 2015.

[79] Bernd Resch, Günther Sagl, Tobias Törnros, Andreas Bachmaier, Jan-Bleicke Eggers, Sebastian Herkel, Sattaya Narmsara, and Hartmut Gündra. GIS-based planning and modeling for renewable energy: Challenges and future research avenues. *ISPRS IJGI*, 3(2):662–692, 2014.

[80] Matej Brumenm, Niko Lukac, and Borut Zalik. GIS application for solar potential estimation on buildings roofs. In *IARIA WEB*, 2015.

[81] Bill Marion and Mary Anderberg. PVWATTS - an online performance calculator for grid-connected PV systems. In *ASES SOLAR*, pages 119–124, 2000.

[82] PVGIS. http://re.jrc.ec.europa.eu/pvg_tools/en/tools.html.

[83] Mapdwell Solar System. http://www.mapdwell.com.

[84] Luıs de Sousa, Christopher Eykamp, Ulrich Leopold, Olivier Baume, and Christian Braun. iguess-a web based system integrating urban energy planning and assessment modelling for multi-scale spatial decision making. In *iEMSs 2012*, 2012.

[85] Simon Schuffert. An automatic data driven approach to derive photovoltaic-suitable roof surfaces from ALS data. In *IEEE JURSE*, pages 267–270, 2013.

[86] F. Kazhamiaka, Y. Ghiassi-Farrokhfal, S. Keshav, and C. Rosenberg. Comparison of different approaches for solar PV and storage sizing. *IEEE TSUSC*, 2019.

[87] R. Atia and N. Yamada. Sizing and analysis of renewable energy and battery systems in residential microgrids. *IEEE TSG*, 7(3):1204–1213, 2016.

[88] Y. Ghiassi-Farrokhfal, F. Kazhamiaka, C. Rosenberg, and S. Keshav. Optimal design of solar PV farms with storage. *IEEE TSUSE*, 6(4):1586–1593, 2015.

[89] Akash Kumar Shukla, K. Sudhakar, and Prashant Baredar. Design, simulation and economic analysis of standalone roof top solar PV system in India. *Solar Energy*, 136:437 – 449, 2016.

[90] M. Severini, A. Scorrano, S. Squartini, M. Fagiani, and F. Piazza. SW framework for simulation and evaluation of partial shading effects in configurable PV systems. In *IEEE EEEIC*, pages 1–6, 2016.

[91] P. Manganiello, M. Baka, H. Goverde, T. Borgers, J. Govaerts, A. van der Heide, E. Voroshazi, and F. Catthoor. A bottom-up energy simulation framework to accurately compare PV module topologies under non-uniform and dynamic operating conditions. In *IEEE PVSC*, pages 3343–3347, 2017.

[92] Xiangyang Gong and Manohar Kulkarni. Design optimization of a large scale rooftop photovoltaic system. *Solar Energy*, 78(3):362 – 374, 2005.

[93] S. Kucuksari, A. M. Khaleghi, M. Hamidi, Y. Zhang, F. Szidarovszky, G. Bayraksan, and Y.-J. Son. An integrated GIS, optimization and simulation framework for optimal PV size and location in campus area environments. *Applied Energy*, 113:1601 – 1613, 2014.

[94] C. Keerthisinghe, G. Verbic, and A. C. Chapman. Evaluation of a multi-stage stochastic optimisation framework for energy management of residential PV-storage systems. In *IEEE AUPEC*, pages 1–6, 2014.

[95] W. Khemiri, R. Yaagoubi, and Y. Miky. Optimal placement of solar photovoltaic farms using analytical hierarchical process and geographic information system in Mekkah, Saudi Arabia. In *Proc. of AIP*, 2018.

[96] Weather Underground. http://www.wunderground.com.

[97] D. J. Pagliari, S. Vinco, E. Macii, and M. Poncino. Irradiance-driven partial reconfiguration of PV panels. In *Proc. of DATE*, pages 884–889, 2019.

[98] Sara Vinco, Enrico Macii, and Massimo Poncino. Optimal topology-aware PV panel floorplanning with hybrid orientation. In *Proc. of ACM GLS-VLSI*, page 491–494, 2018.

[99] Fixr. Solar panel maintenance cost. https://www.fixr.com/costs/solar-panel-maintenance, 2020.

[100] Xiaoxing Lu, Kangping Li, Hanchen Xu, Fei Wang, Zhenyu Zhou, and Yagang Zhang. Fundamentals and business model for resource aggregator of demand response in electricity markets. *Energy*, 204:117885, 2020.

[101] New business model encourages escos to join forces with demand response aggregators. Technical report, European Commission, Executive Agency for SMEs (EASME), Jun 2020.

[102] Li Zhu, Qingxiang Li, Mengdong Chen, Kaiyue Cao, and Yong Sun. A simplified mathematical model for power output predicting of Building Integrated Photovoltaic under partial shading conditions. *Elsevier Energy Conversion and Management*, 180:831–843, 2019.

[103] Mohammad AS Masoum, Seyed M Mousavi Badejani, and Mohsen Kalantar. Optimal placement of hybrid pv-wind systems using genetic algorithm. In *2010 Innovative Smart Grid Technologies (ISGT)*, pages 1–5. IEEE, 2010.

[104] Abigail D Ondeck, Thomas F Edgar, and Michael Baldea. Optimal operation of a residential district-level combined photovoltaic/natural gas power and cooling system. *Applied Energy*, 156:593–606, 2015.

[105] S. Chowdhury and T. Matlokotsi. Optimal placement and sizing of renewable distributed generation in electricity networks considering different load models. In *Proc. of International Universities Power Engineering Conference (UPEC)*, pages 1–6, 2017.

[106] Sadik Kucuksari, Amirreza M Khaleghi, Maryam Hamidi, Ye Zhang, Ferenc Szidarovszky, Guzin Bayraksan, and Young-Jun Son. An integrated gis, optimization and simulation framework for optimal pv size and location in campus area environments. *Applied Energy*, 113:1601–1613, 2014.

[107] Benjamin Pillot, Nadeem Al-Kurdi, Carmen Gervet, and Laurent Linguet. An integrated gis and robust optimization framework for solar pv plant planning scenarios at utility scale. *Applied Energy*, 260:114257, 2020.

[108] Alisa Yushchenko, Andréa De Bono, Bruno Chatenoux, Martin Kumar Patel, and Nicolas Ray. GIS-based assessment of photovoltaic (PV) and concentrated solar power (CSP) generation potential in West Africa. *Renewable and Sustainable Energy Reviews*, 81:2088–2103, 2018.

[109] Nóra Hegedűsné Baranyai, Henrik Zsiborács, András Vincze, Nóra Rodek, Martina Makai, Gábor Pintér, et al. Correlation analysis of the spread of household-sized photovoltaic power plants and various district indicators: A case study. *Sustainability*, 13(2):1–1, 2021.

[110] W Khemiri, R Yaagoubi, and Y Miky. Optimal placement of solar photovoltaic farms using analytical hierarchical process and geographic information system in mekkah, saudi arabia. In *AIP Conference Proceedings*, volume 2056, page 020025. AIP Publishing LLC, 2018.

[111] Luca Bergamasco and Pietro Asinari. Scalable methodology for the photovoltaic solar energy potential assessment based on available roof surface area: Application to piedmont region (italy). *Solar energy*, 85(5):1041–1055, 2011.

[112] Stefano Bracco, Federico Delfino, Giulio Ferro, Luisa Pagnini, Michela Robba, and Mansueto Rossi. Energy planning of sustainable districts: Towards the exploitation of small size intermittent renewables in urban areas. *Applied Energy*, 228:2288–2297, 2018.

[113] Rawad El Kontar and Xin Jin. Optimal efficiency and operational cost savings: A framework for automated rooftop pv placement. In *Proceedings of 2020 Building Performance Analysis Conference and SimBuild, ASHRAE and IBPSA-USA*, 2020.

[114] Electricity price statistics - Statistics Explained. https://ec.europa.eu/eurostat/statistics-explained/index.php/Electricity_price_statistics.

[115] International Renewable Energy Agency. Future of solar photovoltaic: deployment, investment, technology, grid integration and socio-economic aspects. *A Global Energy Transformation*, 2019.

[116] Om Prakash Mahela, Abdul Gafoor Shaik, Neeraj Gupta, Mahdi Khosravy, Baseem Khan, Hassan Haes Alhelou, and Sanjeevikumar Padmanaban. Recognition of power quality issues associated with grid integrated solar photovoltaic plant in experimental framework. *IEEE Systems Journal*, 15(3):3740–3748, 2020.

[117] Conference of the parties (cop) | unfccc.

[118] Hui Dou, Yong Qi, Wei Wei, and Houbing Song. Carbon-aware electricity cost minimization for sustainable data centers. *IEEE Transactions on Sustainable Computing*, 2(2):211–223, 2017.

[119] G. Boluk and R. Kaplan. Effectiveness of renewable energy incentives on sustainability: evidence from dynamic panel data analysis for the EU countries and Turkey. *Environmental Science and Pollution Research*, 29:26613–26630, 2022.

[120] Suraj Suman and Swades De. Low complexity dimensioning of sustainable solar-enabled systems: A case of base station. *IEEE Transactions on Sustainable Computing*, 5(3):438–454, 2020.

[121] Ayhan Demirbas. Global renewable energy projections. *Energy Sources, Part B*, 4(2):212–224, 2009.

[122] Daniel M Kammen and Deborah A Sunter. City-integrated renewable energy for urban sustainability. *Science*, 352(6288):922–928, 2016.

[123] Saulė Milčiuvienė, Julija Kiršienė, Enrique Doheijo, Rolandas Urbonas, and Darius Milčius. The role of renewable energy prosumers in implementing energy justice theory. *Sustainability*, 11(19):5286, 2019.

[124] Marcella Nicolini and Massimo Tavoni. Are renewable energy subsidies effective? Evidence from Europe. *Renewable and Sustainable Energy Reviews*, 74:412–423, 2017.

[125] Future of solar photovoltaic - deployment, investment, technology, grid integration and socio-economic aspects. Technical report, IRENA - International Renewable Energy Agency, 2019.

[126] Daniele Salvatore Schiera, Francesco Demetrio Minuto, Lorenzo Bottaccioli, Romano Borchiellini, and Andrea Lanzini. Analysis of rooftop photovoltaics diffusion in energy community buildings by a novel gis-and agent-based modeling co-simulation platform. *IEEE Access*, 7:93404–93432, 2019.

[127] Johannes Radl, Andreas Fleischhacker, Frida Huglen Revheim, Georg Lettner, and Hans Auer. Comparison of profitability of PV electricity sharing in renewable energy communities in selected european countries. *Energies*, 13(19):5007, 2020.

[128] Energy communities.

[129] Community-ownership models: Innovation landscape brief. Technical report, International Renewable Energy Agency (IRENA), 2020.

[130] Matteo Orlando, Lorenzo Bottaccioli, Sara Vinco, Enrico Macii, Massimo Poncino, and Edoardo Patti. Design of district-level photovoltaic installations for optimal power production and economic benefit. In *2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 1873–1878. IEEE, 2021.

[131] Matteo Orlando, Lorenzo Bottaccioli, Edoardo Patti, Enrico Macii, Sara Vinco, and Massimo Poncino. Optimal configuration and placement of pv systems in building roofs with cost analysis. In *2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 1411–1416. IEEE, 2020.

[132] Sahar Shafiei and Ruhul A Salim. Non-renewable and renewable energy consumption and CO2 emissions in OECD countries: a comparative analysis. *Energy Policy*, 66:547–556, 2014.

[133] Istat.it - 15° censimento della popolazione e delle abitazioni 2011.

[134] Sara Vinco, Daniele Jahier Pagliari, Lorenzo Bottaccioli, Edoardo Patti, Enrico Macii, and Massimo Poncino. A microservices-based framework for smart design and optimization of pv installations. *IEEE Transactions on Sustainable Computing*, 6(4):531–543, 2020.

[135] Han Wang, Nicholas Good, Pierluigi Mancarella, and Kerry Lintern. Pv-battery community energy systems: Economic, energy independence and network deferral analysis. In *2017 14th International Conference on the European Energy Market (EEM)*, pages 1–5. IEEE, 2017.

[136] Antun Pfeifer, Viktorija Dobravec, Luka Pavlinek, Goran Krajačić, and Neven Duić. Integration of renewable energy and demand response technologies in interconnected energy systems. *Energy*, 161:447–455, 2018.

[137] Xiaonan Wang, Ahmet Palazoglu, and Nael H El-Farra. Operation of residential hybrid renewable energy systems: Integrating forecasting, optimization and demand response. In *2014 American Control Conference*, pages 5043–5048. IEEE, 2014.

[138] Hasan Doagou-Mojarrad, GB Gharehpetian, H Rastegar, and Javad Olamaei. Optimal placement and sizing of dg (distributed generation) units in distribution networks by novel hybrid evolutionary algorithm. *Energy*, 54:129–138, 2013.

[139] Adel A Abou El-Ela, Ragab A El-Sehiemy, and Ahmed Samir Abbas. Optimal placement and sizing of distributed generation and capacitor banks in distribution systems using water cycle algorithm. *IEEE Systems Journal*, 12(4):3629–3636, 2018.

[140] Luca Bergamasco and Pietro Asinari. Scalable methodology for the photovoltaic solar energy potential assessment based on available roof surface area: Further improvements by ortho-image analysis and application to turin (italy). *Solar Energy*, 85:2741–2756, 11 2011.

[141] Yan wei Sun, Angela Hof, Run Wang, Jian Liu, Yan jie Lin, and De wei Yang. GIS-based approach for potential analysis of solar PV generation at the regional scale: A case study of fujian province. *Energy Policy*, 58:248–259, 7 2013.

[142] Raul F.C. Miranda, Alexandre Szklo, and Roberto Schaeffer. Technical-economic potential of PV systems on Brazilian rooftops. *Renewable Energy*, 75:694–713, 3 2015.

[143] Brad Guanqiao Huang, Fiodar Kazhamiaka, and Srinivasan Keshav. Sizing solar panels and storage for multiple roofs. In *Proceedings of the Twelfth ACM International Conference on Future Energy Systems*, pages 84–94, 2021.

[144] David A. Jacques, James Gooding, Jannik J. Giesekam, Alison S. Tomlin, and Rolf Crook. Methodology for the assessment of PV capacity over a city region using low-resolution LiDAR data and application to the city of Leeds (UK). *Applied Energy*, 124:28–34, 7 2014.

[145] DJ Damiri, S Legino, and S Amboro. Engineering design development of 52, 5 kilowatt peak solar photovoltaic system for industrial rooftop building. In *Journal of Physics: Conference Series*, volume 1402, page 033087. IOP Publishing, 2019.

[146] S. Vinco, L. Bottaccioli, E. Patti, A. Acquaviva, E. Macii, and M. Poncino. GIS-based optimal photovoltaic panel floorplanning for residential installations. In *Proc. of Design, Automation Test in Europe Conference Exhibition (DATE) 2018*, pages 437–442, 2018.

[147] A Cielo, P Margiaria, P Lazzeroni, I Mariuzzo, and M Repetto. Renewable energy communities business models under the 2020 Italian regulation. *Journal of Cleaner Production*, 316:128217, 2021.

[148] Moiz Masood Syed, Paula Hansen, and Gregory M. Morrison. Performance of a shared solar and battery storage system in an Australian apartment building. *Energy and Buildings*, 225:110321, 10 2020.

[149] Bernadette Fina, Hans Auer, and Werner Friedl. Profitability of PV sharing in energy communities: Use cases for different settlement patterns. *Energy*, 189:116148, 2019.

[150] Fiodar Kazhamiaka, Yashar Ghiassi-Farrokhfal, Srinivasan Keshav, and Catherine Rosenberg. Comparison of different approaches for solar PV and storage sizing; comparison of different approaches for solar PV and storage sizing. *IEEE Transactions on Sustainable Computing*, pages 2377–3782, 2019.

[151] Hadia Awad and Mustafa Gül. Optimisation of community shared solar application in energy efficient communities. *Sustainable cities and society*, 43:221–237, 2018.

[152] Sara Vinco, Lorenzo Bottaccioli, Edoardo Patti, Andrea Acquaviva, and Massimo Poncino. A compact PV panel model for cyber-physical systems in smart cities. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5, 2018.

[153] Midori | free to innovate.

[154] G. Jenks. The data model concept in statistical mapping. Technical report, University of Kansas, 1967.

[155] Greenhouse gas emission intensity of electricity generation in Europe.

[156] Greenhouse gas emissions from a typical passenger vehicle - US EPA.

[157] Uthayasankar Sivarajah, Muhammad Mustafa Kamal, Zahir Irani, and Vishanth Weerakkody. Critical analysis of big data challenges and analytical methods. *Journal of business research*, 70:263–286, 2017.

[158] Fei Hu, Mengchao Xu, Jingchao Yang, Yanshou Liang, Kejin Cui, Michael M Little, Christopher S Lynnes, Daniel Q Duffy, and Chaowei Yang. Evaluating the open source data containers for handling big geospatial raster data. *ISPRS International Journal of Geo-Information*, 7(4):144, 2018.

[159]

[160] European Commission. Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions Powering a Climate-Neutral Economy: An Eu Strategy for Energy System Integration.

[161] European Commission. Powering a climate-neutral economy: An EU Strategy for Energy System Integration, COM/2020/299 final, 2020.

[162] Mike Vogt, Frank Marten, and Martin Braun. A survey and statistical analysis of smart grid co-simulations. *Applied Energy*, 222:67–78, 2018.

[163] Hans-Kristian Ringkjøb, Peter M Haugan, and Ida Marie Solbrekke. A review of modelling tools for energy and electricity systems with large shares of variable renewables. *Renewable and Sustainable Energy Reviews*, 96:440–459, 2018.

[164] Cláudio Gomes, Casper Thule, David Broman, Peter Gorm Larsen, and Hans Vangheluwe. Co-simulation: A survey. *ACM Computing Surveys*, 51(3):49:1–49:33, 2018.

[165] Henrik Lund. Energyplan-advanced energy systems analysis computer model. *Documentation version*, 12:12, 2015.

[166] Peter Lilienthal. Homer® micropower optimization model. Technical report, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2005.

[167] Distributed energy resources customer adoption model plus (DER-CAM+).

[168] Roel De Coninck, Ruben Baetens, Bart Verbruggen, Dirk Saelens, Johan Driesen, and Lieve Helsen. Modelling and simulation of a grid connected photovoltaic heat pump system with thermal energy storage using modelica. In *In Proc. of system simulation*, page 21, 2010.

[169] Jörg Huber and Christoph Nytsch-Geusen. Development of modeling and simulation strategies for large-scale urban districts. In *In Proc. of Building Simulation*, volume 49, 2011.

[170] OFFIS - Institute for Information Technology. Mosaik. https://mosaik.offis.de/.

[171] Ishtiaq Ahmad, Jawad Haider Kazmi, Mohsin Shahzad, Peter Palensky, and Wolfgang Gawlik. Co-simulation framework based on power system, AI and communication tools for evaluating smart grid applications. In *In Proc. of IEEE SGT ASIA*, pages 1–6, 2015.

[172] Florian Schloegl, Sebastian Rohjans, Sebastian Lehnhoff, Jorge Velasquez, Cornelius Steinbrink, and Peter Palensky. Towards a classification scheme for co-simulation approaches in energy systems. In *In Proc. of EDST*, pages 516–521. IEEE, 2015.

[173] Dhananjay Bhor, Kavinkadhirselvan Angappan, and Krishna M Sivalingam. Network and power-grid co-simulation framework for smart grid wide-area monitoring networks. *Journal of Network and Computer Applications*, 59:274–284, 2016.

[174] Luca Barbierato, Enrico Pons, Andrea Mazza, Ettore Francesco Bompard, Vetrievel Subramaniam Rajkumar, Peter Palensky, Enrico Macii, Lorenzo Bottaccioli, and Edoardo Patti. Stability and accuracy analysis of a distributed digital real-time cosimulation infrastructure. *IEEE Tran. on Industry Applications*, 58(3):3193–3204, 2022.

[175] Modelica Association Project. Functional Mockup Interface. https://fmi-standard.org/.

[176] Vincent Reinbold, Christina Protopapadaki, Jean-Philippe Tavella, and Dirk Saelens. Assessing scalability of a low-voltage distribution grid co-simulation through functional mock-up interface. *Journal of Building Performance Simulation*, pages 1–13, 2019.

[177] Yuanda Gao, Desen Kirli, Mehdi Zeinali, Shubhankan Mukherjee, Aziza Birzhanova, John Thompson, Naran Pindoriya, and Aristides Kiprakis. Development of a hardware in-the-loop co-simulation platform for smart distribution networks. In *In Proc. of EVER*, pages 1–10. IEEE, 2020.

[178] Tung Lam Nguyen, Quoc Tuan Tran, Yvon Besanger, et al. Synchronization conditions and real-time constraints in co-simulation and hardware-in-the-loop techniques for cyber–physical energy system assessment. *Sustainable energy, grids and networks*, 20:100252, 2019.

[179] Ferdinanda Ponci, Antonello Monti, and Andrea Benigni. Simulation for the design of smart grid controls. In *In Proc. of IEEE SGMS*, pages 73–78, 2011.

[180] Steffen Vogel, Vetrivel Subramaniam Rajkumar, Ha Thi Nguyen, Marija Stevic, Rishabh Bhandia, Kai Heussen, Peter Palensky, and Antonello Monti. Improvements to the co-simulation interface for geographically distributed real-time simulation. In *In Proc. of IECON*, volume 1, pages 6655–6662, 2019.

[181] Antonello Monti, Marija Stevic, Steffen Vogel, Rik W. De Doncker, Ettore Bompard, Abouzar Estebsari, Francesco Profumo, Rob Hovsapian, Manish Mohanpurkar, Jack David Flicker, Vahan Gevorgian, Siddharth Suryanarayanan, Anurag K. Srivastava, and Andrea Benigni. A global real-time superlab: Enabling high penetration of power electronics in the electric grid. *IEEE Power Electronics Magazine*, 5(3):35–44, 2018.

[182] Abouzar Estebsari, Matteo Orlando, Enrico Pons, Andrea Acquaviva, and Edoardo Patti. A novel internet-of-things infrastructure to support self-healing distribution systems. In *2018 International Conference on Smart Energy Systems and Technologies (SEST)*, pages 1–6. IEEE, 2018.

[183] CEN CENELEC. CEN CENELEC EN 50160 Voltage characteristics of electricity supplied by public distribution systems. https://www.se.com/ww/library/SCHNEIDER_ELECTRIC/SE_LOCAL/APS/204836_1312/DraftStandard0026rev2-DraftEN501602005-05.pdf.

[184] Matteo Orlando, Abouzar Estebsari, Enrico Pons, Marco Pau, Stefano Quer, Massimo Poncino, Lorenzo Bottaccioli, and Edoardo Patti. A smart meter infrastructure for smart grid iot applications. *IEEE Internet of Things Journal*, 2021.

[185] Daniele Salvatore Schiera, Luca Barbierato, Andrea Lanzini, Romano Borchiellini, Enrico Pons, Ettore Bompard, Edoardo Patti, Enrico Macii, and Lorenzo Bottaccioli. A distributed multimodel platform to cosimulate

multienergy systems in smart buildings. *IEEE Transactions on Industry Applications*, 57(5):4428–4440, 2021.

[186] D. S. Schiera, L. Barbierato, A. Lanzini, R. Borchiellini, E. Pons, E. F. Bompard, E. Patti, E. Macii, and L. Bottaccioli. A distributed platform for multi-modelling co-simulations of smart building energy behaviour. In *In Proceedings of 2020 IEEE EEEIC / I&CPS Europe*, pages 1–6, 2020.

Appendix A

List of publications

A.1 International Journals

- *A Resources Ecosystem for Digital and Heritage-Led Holistic Knowledge in Rural Regeneration* Tamborrino, Rosa Rita Maria, Edoardo Patti, Alessandro Aliberti, Mesut Dinler, **Matteo Orlando**, Claudia de Luca, Simona Tondelli, et al. in JOURNAL OF CULTURAL HERITAGE 57 (2022) 265–75. https://doi.org/10.1016/j.culher.2022.09.012.

- *Engaging Users in Resource Ecosystem Building for Local Heritage-Led Knowledge* Tamborrino, Rosa Rita Maria, M. Dinler, E. Patti, A. Aliberti, **M. Orlando**, C. De Luca, S. Tondelli, Z. Amirzada, and I. Pavlova in SUSTAINABILITY 14, no. 8 (2022). https://doi.org/10.3390/su14084575.

- *A Smart Meter Infrastructure for Smart Grid IoT Applications* **Matteo Orlando**, Abouzar Estebsari, Enrico Pons, Marco Pau, Stefano Quer, Massimo Poncino, Lorenzo Bottaccioli, and Edoardo Patti, in IEEE INTERNET OF THINGS JOURNAL 9, no. 14 (2022): 12529–41. https://doi.org/10.1109/JIOT.2021.3137596.

- * *A framework for economic and environmental benefit through Renewable Energy Community* **Matteo Orlando**, Lorenzo Bottaccioli, Stefano Quer, Massimo Poncino, Sara Vinco and Edoardo Patti in IEEE Systems Journal - UNDER REVIEW

- * *A Hybrid Multi-Model Co-Simulation Infrastructure to foster Energy Systems Integration* Luca Barbierato, Daniele Salvatore Schiera, **Matteo Orlando**, Andrea Lanzini, Enrico Pons, Lorenzo Bottaccioli, and Edoardo Pattiin IEEE Transactions on Sustainable Computing - UNDER REVIEW

A.2 International Conferences

- *Design of District-Level Photovoltaic Installations for Optimal Power Production and Economic Benefit* **Matteo Orlando**, Lorenzo Bottaccioli, Sara Vinco, Enrico Macii, Massimo Poncino, and Edoardo Patti, in Proceedings of the IEEE International Computer Software and Applications Conference (COMPSAC). IEEE, 2021. https://doi.org/10.1109/COMPSAC51774.2021.00283.
- *Optimal Configuration and Placement of PV Systems in Building Roofs with Cost Analysis* **Matteo Orlando**, Lorenzo Bottaccioli, Edoardo Patti, Enrico Macii, Sara Vinco, and Massimo Poncino in Proceedings of 2020 IEEE International Computer Software and Applications Conference (COMPSAC), 1411–16. IEEE, 2020. https://doi.org/10.1109/COMPSAC48688.2020.00-58.
- *A Novel Internet-of-Things Infrastructure to Support Self-Healing Distribution Systems* Estebsari, Abouzar, **Matteo Orlando**, Enrico Pons, Andrea Acquaviva, and Edoardo Patti. In Proceedings of International Conferece on Smart Energy Systems and Technologies (SEST 2018), 1–6. IEEE, 2018. https://doi.org/10.1109/SEST.2018.8495717.

A.3 Others

Data-Driven Predictive Maintenance: A Methodology Primer Cerquitelli, Tania, Nikolaos Nikolakis, Lia Morra, Andrea Bellagarda, **Matteo Orlando**, Riku Salokangas, Olli Saarela, Jani Hietala, Petri Kaarmila, and Enrico Macii, in Predictive Maintenance in Smart Factories, 39–73. Springer, 2021. https://doi.org/10.1007/978-981-16-2940-2_3.