

A Real-Time Based Platform for Integrating Power-to-Gas in Electrical Distribution Grids

Original

A Real-Time Based Platform for Integrating Power-to-Gas in Electrical Distribution Grids / Diaz-Londono, C., Fambri, G., Mazza, A., Badami, M., Bompard, E.. - ELETTRONICO. - (2020), pp. 1-6. (55th International Universities Power Engineering Conference, UPEC 2020 Torino (Italia) 1-4 Sept. 2020) [10.1109/UPEC49904.2020.9209803].

Availability:

This version is available at: 11583/2854264 since: 2020-12-01T09:26:41Z

Publisher:

Institute of Electrical and Electronics Engineers Inc.

Published

DOI:10.1109/UPEC49904.2020.9209803

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

IEEE postprint/Author's Accepted Manuscript

©2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

An Electrical Distribution Grid Simulation Tool for Planning Power-to-Gas Integration

Cesar Diaz-Londono*, Gabriele Fambri, Andrea Mazza, Marco
Badami, Ettore Bompard

Dip. Energia "Galileo Ferraris" Politecnico di Torino
Torino, Italy

{cesar.diazlondono, gabriele.fambri, andrea.mazza, marco.badami,
ettore.bompard}@polito.it

ABSTRACT

The high penetration of renewable energy sources in electrical distribution grids gives rise to challenges in the grid balance. Reverse power flow is one of the main issues of non-proper network operation. Currently, this issue is solved by disconnecting the renewable generation, wasting important energy production. In this article, an electrical distribution grid simulation tool is introduced with the purpose of integrating energy conversion systems capable to absorb renewable energy. The power-to-gas technology is considered with a straightforward model and its power demand is defined by a control algorithm. The electrical grid simulation technicalities are presented and implemented in a real-time simulation platform. The automatic creation of case studies considering an external code is developed for minimum and simple user actions. Finally, a case study is carried out considering a scenario without conversion systems and another considering power-to-gas technology. The results present a reverse power flow reduction of 91% and a reduction in electrical grid losses.

Keywords

Renewable Energy Integration, Balance Load Optimization, Power-to-Gas, RT-Lab Simulation Tool, Automatic Cases Creation.

1 Introduction

The high penetration of Renewable Energy Sources (RES) in electrical distribution grids is leading to problems with grid stability.

The existing network and consumption infrastructure is not designed for managing RES variations. Therefore, an increase in flexibility is required by the system operator.

A methodological approach for the analysis of the synergies between different energy networks in order to cope with the increasing RES penetration is presented in [1].

A specific definition of flexibility to be used for an electric vehicle charging station is provided in [2].

In [3], an aggregation strategy of thermoelectric refrigeration units is proposed to offer downward and upward flexibility in energy consumption.

The planning and operational tools for optimizing energy flows and synergies between energy networks project, i.e., the PLANET project [4] aims to design and develop a Decision Support System (DSS) for grid operational planning and management to identify and assess optimal strategies. The purpose is to deploy, integrate and operate conversion systems on the distribution network, considering future energy system scenarios. In [5] how the PLANET DSS tool evaluates, manages, and dispatches the flexibility of Power to Gas/Heat (P2X) technologies is presented.

Therefore, in this paper, a real-time simulation of an electrical distribution grid is presented as a software tool for planning and operating Power-to-Gas (P2G) conversion systems, considering RES as well as the uncontrollable demands. A control algorithm is developed with the purpose of defining the power demand set-point of the conversion systems.

The rest of the paper is organized as follows: Section 2 describes the real-time simulation implementation. In section 3, the main components of the simulation tool are presented. In Section 4, the guidelines for creating automatic study cases is developed. Section 5 introduces the case study and the simulation results. Finally, conclusions and future work are reported in Section 6.

2 Real-time simulation Description

This section aims to present a general description of the real-time simulation. It is performed in the international laboratory for real-time simulation *G-RTS Lab*, at *Politecnico di Torino*. The facility used is composed of a real-time digital simulator (OPAL-RT 5600, see Figure 1) with 7 activated cores and the possibility to perform simulations with the eMEGASIM platform, allowing to perform ElectroMagnetic Transient (EMT) simulations with networks composed of 300 three-phase nodes.



Figure 1: Real-time simulator.

In order to set up the simulation environment, the software RT-LAB which is the built-on software of OPAL-RT Technologies is used. RT-LAB is a real-time simulation platform for high fidelity plant simulation, control system prototyping, and embedded data acquisition and control. Besides, RT-LAB implements the real-time simulation of models directly from Simulink. Therefore, the models are implemented in Simulink following a system simulation and visualization of the system status, which considers the results of its subsystems; then, two subsystems are needed, the Subsystem of the Console (SC) and the Subsystem of the Master (SM). Figure 2 shows this environment, considering the two additional blocks. First, a Model Initialization block used to charge all the parameters of the models and the time step of the simulation. Note that these parameters are described in a MATLAB code. Second, the library *powergui* required block for simulating the Simscape Power Systems Specialized Technology models used in the electrical grid model. This library allows choosing different methods to solve the circuit; however, in this work, a discrete solver is considered, the solution is carried out at fixed time steps.

2.1 Console subsystem

The console subsystem presents the user interface, see Figure 3. This is the only available window for the user during the execution of the simulation in real-time. Therefore, the main output variables are depicted, such as the total power consumption of the electrical grid, the voltage in the nodes, and the energy consumed by the energy conversion systems (assets). The implemented blocks in the console interface are:

- *OpComm*: This block is used to communicate in real-time the subsystems SC and the SM; then, any signal entering the console or master subsystem must go through the OpComm block.
- *Visualization*: It is developed through the cyan blocks presented in Figure 3, which display the generation of the signals during the simulation. These blocks depict time-domain signals with respect to the simulation time, i.e., the real-time fixed in the simulation.

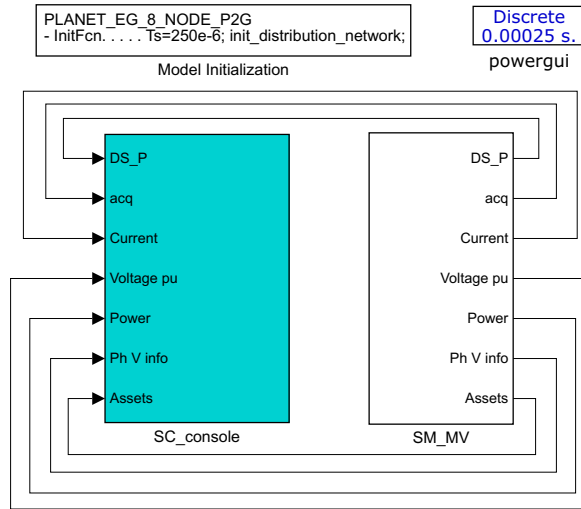


Figure 2: Model considering Console (SC) and Master (MV) subsystems.

- *Monitoring*: This information refers to the data that is saved in the MATLAB workspace and the following three events:
 - Computation time: corresponds to the real-time spent in the execution of each time step.
 - Real-time size: corresponds to the total time length (computing + overhead + synchronization) of execution of each time step.
 - Numbers of overruns: corresponds to the number of overruns detected during the execution of the complete model at each time step.

2.2 Master subsystem

Inside the master subsystem, the different interconnections between the components of the simulation are presented, i.e., where all computational elements, mathematical operations, and signal generators are implemented. Figure 4 presents the integration of the following components:

- *Electrical grid*: an 8-node distribution electrical grid is considered, which receives the aggregated electrical demand and renewable generation profiles in each node.
- *Uncontrollable nodal loads and renewable profiles*: data of residential, industrial, and commercial users as well as a photo-voltaic generation are provided every 15 minutes to each node of the electrical grid.

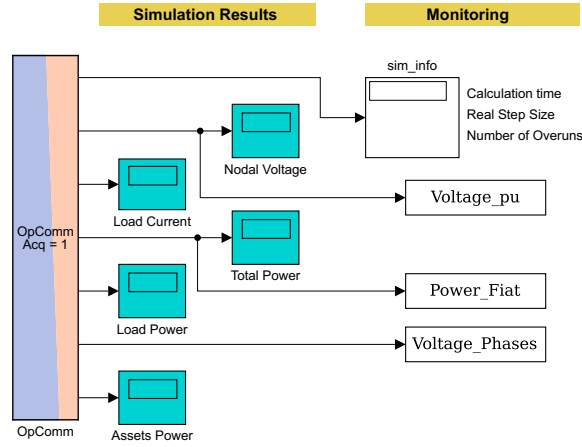


Figure 3: Real-time simulation console subsystem.

- *Conversion systems*: the Power-to-Gas (P2G) asset is taken into account, which produces Synthetic Natural Gas (SNG) to store electricity.
- *Control Algorithm*: the controller is entrusted for controlling the P2G energy consumption by defining the operation set-point at each time step.
- *Monitoring* (not presented in Figure 4): contains the OpMonitor RT-LAB block that provides information on the model timing (Computation time, Real-time size, and Numbers of overruns).
- *Data save* (not presented in Figure 4): carried out with the OpWrite-File RT-LAB block, with the purpose of saving the master signals such as voltage, power, and current.

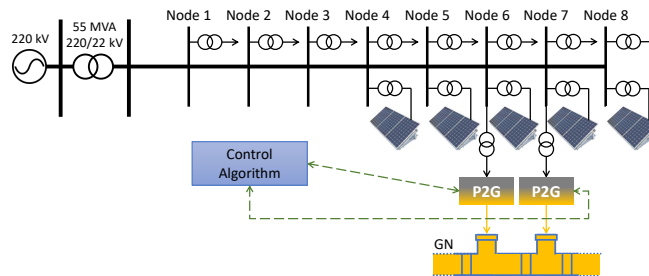


Figure 4: Simulation implementation scheme in the master subsystem.

Therefore, in the SM, the electrical grid is feed with the aggregated consumption (considering the passive demand and the P2G systems demands)

and the renewable generation patterns. These profiles consider active and reactive power. Moreover, P2G systems report the flexibility they have at each step, subsequently, all these data are sent to the control algorithm, which calculates the operation set point of the P2G components, considering the reported flexibilities and the power consumption and renewable generation. In addition, the P2G systems are connected to nodes 6 and 7. It is worth to mention that no Gas Network (GN) model is considered at the output (SNG) of the P2G model. The GN is supposed to be with no constraints, i.e., the gas generated by the P2G can be delivered at any moment and with any amount.

3 Integration of the Simulation Components

This section aims to present the considered models in the SM subsystem. Therefore, an electrical distribution grid, a straightforward P2G model, and the control algorithm are introduced.

3.1 Electrical Distribution Grid

The electric grid considered in this work is part of a Medium Voltage (MV) network with five feeders derived from three 22 kV bus-bars of a 220/22 kV primary substation at a frequency of 50 Hz. Therefore, the analysis is developed taking into account the 8-node grid presented in Figure 4.

3.1.1 Main Electrical Grid

Figure 5 presents the implemented electrical grid, where the principal elements are:

- Conventional generation: The complete grid is fed by a *Three-Phase Voltage Source* with 220 kV and 50 Hz which is connected in series to a *Three-Phase RLC Branch block*.
- Measurement System: Measurement blocks (inside the cyan color blocks) are considered. First, two *Three-Phase V-I Measurements* are used to measure instantaneous voltages and currents in the generation, it is considered one for phase-to-ground voltage and other for phase-to-phase voltage measurement. Second, a three-phase active power (P) and reactive power (Q) block is implemented, it is associated with a periodic set of three-phase voltages and currents to measure P and Q , i.e., the *Three-Phase V-I Measurement* output is the input for computing P and Q . Hence, these blocks allow the following measurements:
 - Phase Voltage.
 - Instantaneous current.

- Instantaneous voltage.
 - Voltage p.u. (per unit), base voltage $V_b = 220$ kV.
 - Active power.
 - Reactive power.
- Transformer. It is implemented considering a *three single-phase transformer* from 220 kV to 22 kV (55 MVA). The two windings are connected in delta at the input and in Y at the output.
 - 8-Node Grid. The 8 nodes are inside the green block which not only takes into account the three-phases but also the load demand (LP) and renewable generation (GP) profiles.

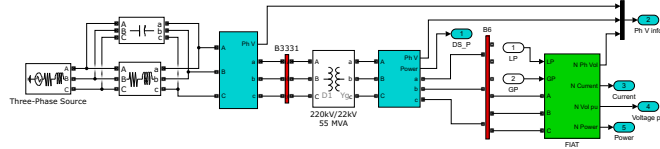


Figure 5: Main electrical grid.

3.1.2 8-node Electrical Grid.

In Figure 6, the 8-node grid Simulink implementation is shown, where the nodes are depicted in red bars connected through transmission lines. These lines (white blocks) are implemented as a balanced three-phase transmission line model with parameters lumped in a PI section. In particular, the first node is located 2.8 km from the transformer; whereas, the transmission lines among the nodes have distances between 233 m and 416 m. Moreover, the orange blocks collect the consumption and renewable generation profiles that must be followed by each node; whereas, both profiles are provided in the gray blocks.

In addition, each gray block has two *Three-Phase Dynamic Load* blocks with the purpose of simulating the load consumption and the renewable generation. These blocks receive as input the active P and reactive Q power that varies in time. In fact, the renewable generation is implemented as a negative load. Moreover, each node has the measurement systems (described previously) for monitoring all energy variations and for control purposes.

3.2 Power-to-Gas Conversion Systems

The main components of a P2G plant are the electrolyzer and the methanation reactor. The electrolyzer converts an electricity input into hydrogen

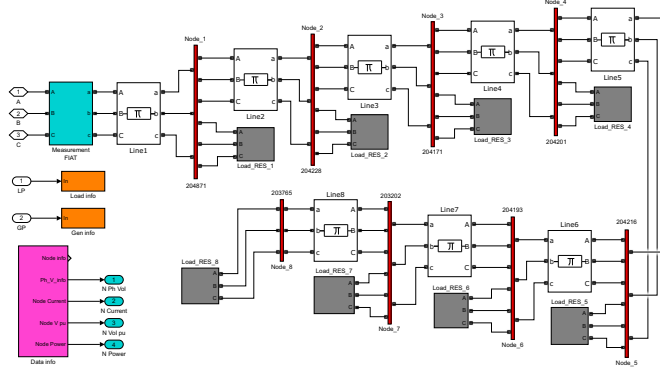


Figure 6: 8-node grid Simulink implementation.

(H2). The hydrogen is then converted into Synthetic Natural Gas (SNG) during the methanation process. These conversion systems are a suitable solution to compensate the renewable production fluctuations, owing to the flexibility that can provide to the electrical grid.

P2G systems can rapidly change their working set-point, Polymer Electrolyte Membrane (PEM) electrolyzers can be completely ramped up and down (0÷100% range) in just a few seconds [6]. The methanation reactors are slower but can they still have good flexibility: e.g., the load of a catalytic reactor can be modified with a ramp rate constraint of 3% in one minute [7]. In this study, it was considered that the P2G system can increase/decrease its load by 30% of its nominal capacity for each simulation step of 15 minutes. The minimum electric load of the P2G plant is assumed to be 5% of its nominal capacity.

The P2G process can be summarized by the following simplified equation:

$$q_{SNG} = \eta_{el} \eta_{meth} u_{P2G}, \quad (1)$$

where q_{SNG} is the SNG production, η_{el} and η_{meth} are the electrolysis and methanation reactor efficiencies, and u_{P2G} is the electric consumption of the P2G plant. For the sake of simplicity it has been considered that the SNG can be injected in the gas network without limitations.

3.3 Control Algorithm

The P2G plants are flexible units as their load can be modified within the ramp rate, minimum and maximum power constraints. This flexibility can be used to improve network parameters.

In this study, the flexibility of P2G is used to absorb the renewables overproduction thus improving the balance of the electricity network. A controller is executed each simulation step (15 minutes), the control algorithm receives as input the total load and generation power in the grid, and

the P2G flexibility defined as the maximum and minimum power that they can absorb at the given timestep. The algorithm regulates the P2G load based on the conditions of the network and the availability of P2G systems: if there is no production of RES overproduction, P2G systems are kept at minimum load, in case of overproduction the P2G load is increased to in order to absorb overproduction.

4 Automatic Creation of Case Studies

In this section, how to set a case study with minimum user actions is explained. Then, the way to create new scenarios based on a predefined layout; and the selection of the load and generation profiles, as well as the P2G systems is presented.

4.1 Case settings without user actions

In order to run the simulation in the OPAL-RT machine through the RT-LAB software in an automatic way, a Python code is developed. This automatic behavior is possible thanks to the RT-LAB API library, which provides a set of functions that allow external programs for interacting with RT-LAB. Therefore, the code is able to:

- *Select the project containing the Simulink model.* To open a project from its settings file is considered the command `OpenProject(ProjectPath)`.
- *Open the SM model.* To load the model, i.e., the block diagram of the current connected model, this is possible with `LoadOriginalModel`. Moreover, the command `ResetOriginalModel` close the Simulink model.
- *Send commands to MATLAB.* To run a MATLAB function that returns a message is needed the function `result = ExecuteMatlabCmd(cmd)`, where `result` is the message returned by MATLAB, and `cmd` is the MATLAB input.
- *Build the model.* To start the compilation, the function `StartCompile2(compilationInfo)` is used. This function separate the subsystems, generates the code and compile each subsystem independently.
- *Load the model into the target.* To load the model's executables in memory, the command `Load(SimulationMode, TimeFactor)` is used. Moreover, to open the Console subsystem and visualized the simulation results the command `LoadConsole` is needed.
- *Start the simulation.* To start the simulation, the commands `ExecuteConsole` and `Execute(TimeFactor)` must be run.

- *Modify parameters.* To set any parameter of the model the following steps are needed:
 - Releases the parameter control of the model. This control allows the user to change the values of models' parameters, and it is possible with the function `GetParameterControl(ParControl)`.
 - Change the value of the parameter using their path (`Name`) and value, considering the function `SetParametersByName(Name, Value)`.

This option of modifying the parameters allows the user to change values, whether of a block or a constant to enable a subsystem. However, not all the block parameters are available; then, an array with the new values is created in the Python code and saved in project folder (this must be done before building the model).

- *Simulation duration.* To set this time, the function `Time.sleep(Timer)` can be used, taking into account the `Timer` in seconds.
- *Reset the simulation.* To stop and reset the current simulation is needed the command `Reset`.

To recapitulate and explain the Python script implemented, Figure 7 depicts the logic followed. In fact, the blue blocks are the ones described previously. The green blocks are added to import the required libraries and define the electrical grid configuration and P2G parameters. Moreover, the MATLAB code (orange block) runs different MATLAB functions not only for modifying the Simulink model configuration and save it as a new model that subsequently is compiled, but also at the end of the simulation to compute the grid results. In addition, the user interface (Figure 3) is available when the SC is opened.

4.2 Creation of new case studies

The creation of new scenarios is carried out based on a predefined net layout. This layout considers a complete electrical grid, i.e., the 8-node grid. Likewise, the maximum number of P2G is eight (one per each node), considering each power and each flexibility capacities. Therefore, the MS of Figure 4 will considers the same electrical grid; however, the P2G systems are contemplated in all the nodes.

The 8-node electrical grid implementation leads to activate or deactivate the nodes, i.e., the user can define the grid configuration (based on this grid). The electrical grid setup is defined with a vector of binary elements, for example:

`Set_EleGrid = (1, 1, 1, 1, 1, 1, 0, 0)`, this means that nodes 7 and 8 are deactivated. Note that this configuration is fixed in the Python

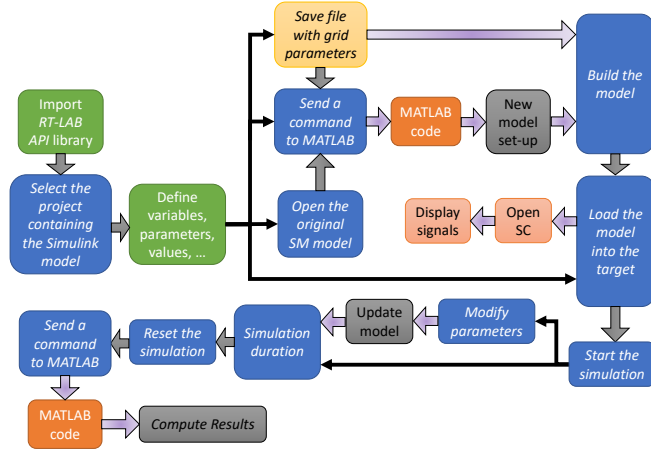


Figure 7: Automatic case study implementation

script and developed with the MATLAB code. The grid configuration is carried out considering the set block parameter tool as `set_param`.

In addition, each P2G has an enable signal which activate or deactivate the system. Thereby, the user can easily configure each node. Note that this information is set in the Python script, considering the vector, for example as:

$\text{Set_P2G} = (1, 0, 0, 1, 0, 1, 0, 0)$, where, each element of the vectors corresponds to the activation/deactivation of the system in a specific node. Moreover, this example is aligned with the electrical grid setting, nodes 7 and 8 are deactivated; therefore, no conversion system can be considered in these nodes.

In such a way, create new case studies based on a predefined layout is not computationally demanding due to the network can be simpler (systems which are not enables, do not run) but not more complex.

5 Case Study

The real-time simulation case study is presented considering the architecture of Figure 4, where all the features are loaded and executed in the Python code (see Section 4). The sample time is 15 minutes and simulation horizon is one day (96 steps).

The PV production is considered in the nodes 4 to 8. In Figure 8 the total PV production is presented, where the total PV production is 21.78 kWh.

The case study considers two scenarios:

- *Scenario 1*: do not consider the conversion systems.

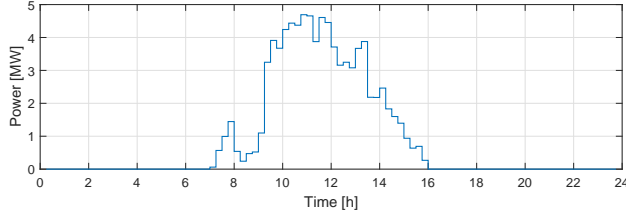


Figure 8: PV production.

- *Scenario 2*: considers the conversion systems, i.e., the P2G systems connected to node 6 and 7.

Table 1 shows the P2G parameter used in the simulation. Moreover, the P2G connected to node 6 has maximum power of 0.8 MW; whereas, the one connected to node 7 is 1 MW.

Table 1: P2G model parameters.

Parameter	Value
Electrolysis efficiency [8]	75%
Methanation reactor efficiency [8]	80%
Minimum load [9]	5%
Maximum load	100%
Ramp up / down [7]	30% (15 min)
Power factor [??]	0.9

The power demand by the grid in all the simulation horizon by both scenarios is presented in Figure 9. Table 2 presents the electrical grid results. There can be seen that in scenario 2 a reduction of 91.0% is presented owing to the P2G systems consumption.

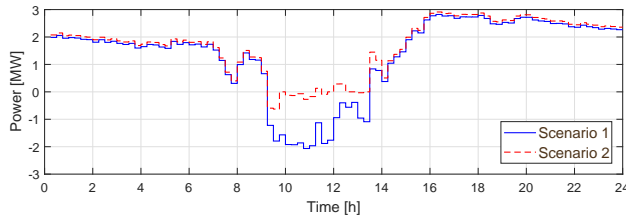


Figure 9: Power in the electrical network, for Scenario 1 and 2.

The P2G power demand in the simulation horizon is presented in Figure 10. It is noticed that the conversion systems consume energy at the minimum power when there is no reverse power flow, while there is an important consumption for reducing the unbalance.

Table 2: Electrical grid results.

Parameter	Scenario 1	Scenario 2
Reverse Power Flow	5.86 MW	0.53 MW
Time of Reverse Power Flow	3 h 45'	2 h 15'
Total consumption	54.28 kWh	61.77 kWh
Grid self-sufficiency	29.3%	39.1%
Grid self-consumption	73.1%	97.6%
Grid losses	183.5 kWh	170.5 kWh
Percentage grid losses	0.34%	0.28%

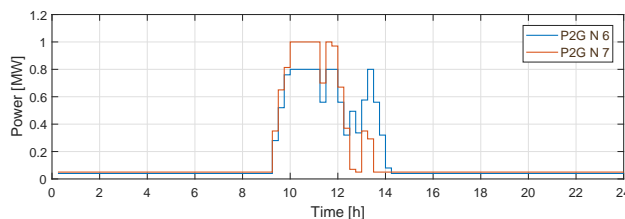


Figure 10: Power-to-Gas power demand.

In addition, Figure 11 and Figure 12, presents the voltage and loading factor response.

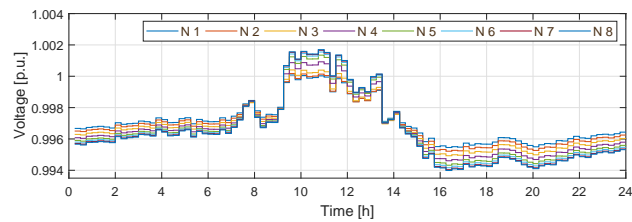
6 Conclusions and future work

Acknowledgments

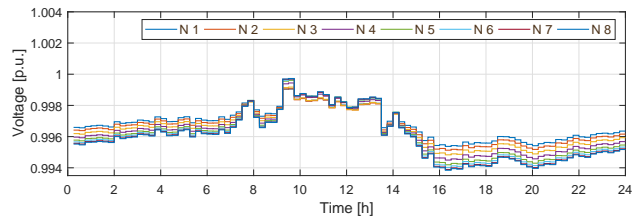
This research received support from the EU project Planet founded from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 773839

References

- [1] M. Badami and G. Fambri, “Optimising energy flows and synergies between energy networks,” *Energy*, vol. 173, pp. 400–412, 2019.
- [2] C. Diaz-Londono, L. Colangelo, F. Ruiz, D. Patino, C. Novara, and G. Chicco, “Optimal Strategy to Exploit the Flexibility of an Electric Vehicle Charging Station,” *Energies*, vol. 12, no. 20, pp. 1–29, 2019.
- [3] C. Diaz-Londono, D. Enescu, F. Ruiz, and A. Mazza, “Experimental Modeling and Aggregation Strategy for Thermoelectric Refrigeration Units as Flexible Loads,” *Applied Energy*, vol. 272, 2020.



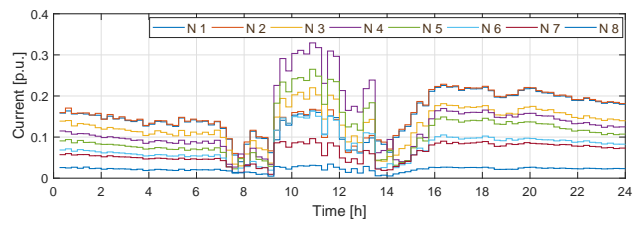
(a) Scenario 1.



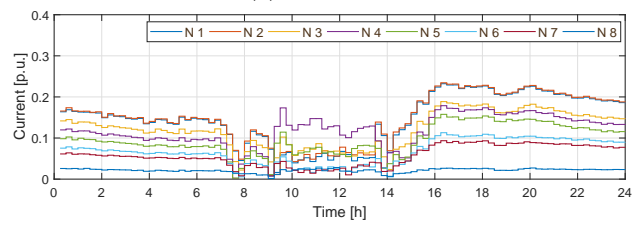
(b) Scenario 2.

Figure 11: Voltage profiles.

- [4] PLANET Project, “<https://www.h2020-planet.eu/>.”
- [5] M. Badami, G. Fambri, S. Mancò, M. Martino, I. G. Damousis, D. Agtzidis, and D. Tzouvaras, “A decision support system tool to manage the flexibility in renewable energy-based power systems,” *Energies*, vol. 13, no. 1, pp. 1 – 16, 2020.
- [6] A. Mazza, E. Bompard, and G. Chicco, “Applications of power to gas technologies in emerging electrical systems,” *Renewable and Sustainable Energy Reviews*, vol. 92, pp. 794–806, 2018.
- [7] F. Mörs, S. Ruth, J. Gorre, and R. Leonhard, “D5.9 final report on evaluation of technologies and processes,” tech. rep., H2020 STORE & GO Project, 2020.
- [8] E. Kötter, L. Schneider, F. Sehnke, K. Ohnmeiss, and R. Schröer, “The future electric power system: Impact of Power-to-Gas by interacting with other renewable energy components,” *Journal of Energy Storage*, vol. 5, pp. 113–119, 2016.
- [9] A. Godula-Jopek, *Hydrogen production: by electrolysis*. 2015.



(a) Scenario 1.



(b) Scenario 2.

Figure 12: Lines loading factor profiles.