

edgeFLEX

D1.3

Dynamic-phasor driven voltage control concept for Energy Communities with future VPPs

The research leading to these results has received funding from the European Union's Horizon 2020 Research and Innovation Programme, under Grant Agreement no 883710.

Project Name	edgeFLEX
Contractual Delivery Date:	31.03.2022
Actual Delivery Date:	31.03.2022
Author(s):	RWTH – UNIBO
Workpackage:	WP1 – Dynamic-phasor Driven Voltage Control Concepts for Dynamically Controlled VPP Solutions
Security:	P
Nature:	R
Version:	V1.0
Total number of pages:	22

Abstract

This deliverable describes the second voltage control algorithm developed in the second reporting period of the project. The algorithm is based on a Model Predictive Control strategy and it integrates an interface with the Local Flexibility Market to engage the customers.

Keyword list

Voltage control, Model Predictive Control, Powerflow, MQTT, VPP

Disclaimer

All information provided reflects the status of the edgeFLEX project at the time of writing and may be subject to change.

Executive Summary

To mitigate voltage violations occurring in the distribution grid during normal operating condition, smart solutions based on controlling the installed Distributed Generators (DGs) can be applied. In D1.2 an online voltage control to compensate for voltage violations has been developed, considering the assets directly controllable by the Distribution System Operators (DSOs). The engaging of the customers could, in this sense, increase the ability of the control to compensate for voltage violations on the grid by using not only the assets owned by the DSO but also those owned by the customer. Moreover, as the voltage violations are a local problem, it is reasonable to try to compensate for it by using all the available resources installed in the local grid.

The object of WP1 in this second reporting period of the project was to develop a voltage control algorithm able to engage the customers as part of an energy community by interfacing with the Local Flexibility Market.

The document provides the description of the algorithm and preliminary results performed on a simulated environment to demonstrate the ability of the control algorithm to interface with the Flexibility Market. The application of the control algorithm allows the customers to participate in the control of the voltage and for the DSO it provides the ability to requests the support of the customers to the Flexibility Market.

Authors

Partner	Name	e-mail
RWTH		
	Edoardo De Din	ededin@eonerc.rwth-aachen.de
	Gianluca Lipari	glipari@eonerc.rwth-aachen.de

Table of Contents

1. Introduction	5
1.1 Objective of the report.....	5
1.2 Outline of the report	5
1.3 How to Read this Document	5
2. Integrating the Market in the control to engage the customers.....	7
3. Model Predictive Control (MPC)	9
3.1 Algorithm Description	9
4. Interaction with the Flexibility Market	12
4.1 Calculation of the Flexibility Previsions.....	12
4.2 Flexibility Market (basic implementation)	12
4.3 MPC integrates the flexibility response.....	13
4.4 Powerflow simulation	13
4.5 Simulation Setup	13
5. Preliminary results.....	15
5.1 Grid under test	15
6. Conclusion	18
7. List of Tables	19
8. List of Figures	20
9. References.....	21
10. List of Abbreviations	22
11. Appendix.....	23
11.1 Grid Data.....	23

1. Introduction

This report is based on the result of the work done in Task 1.3, which deals with the definition and implementation of new dynamic-phasor driven voltage control concepts as an ancillary service for LV Energy Communities.

In Deliverable D1.2 the online voltage control using the distribution grid assets has been presented, which was the result of Phase 1 of the project, from month 1 to month 12 (March 2021). This Deliverable focuses on Phase 2 of the project, which extends from month 13 to month 36 (March 2023). The aim of this second phase is to develop a control strategy capable of engaging customers in the control of the voltage by means of the Local Flexibility Market. Thus, local energy communities can respond to flexibility requests made by the voltage control and support the Distribution System Operator (DSO) to compensate for undesired voltage values.

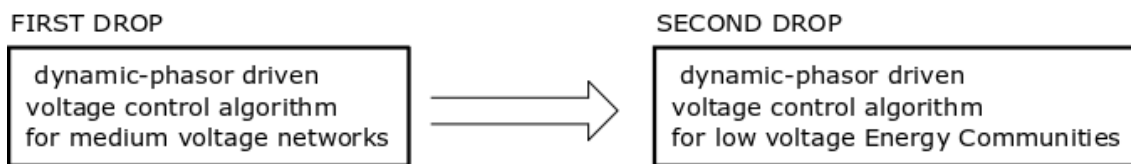


Figure 1: First and Second drops of the voltage control algorithms

To calculate the control output for the Flexibility Market the proposed control algorithm makes use of the phasor measurements obtained by the edgePMU, presented in D4.2, similarly to what has been described in D1.2.

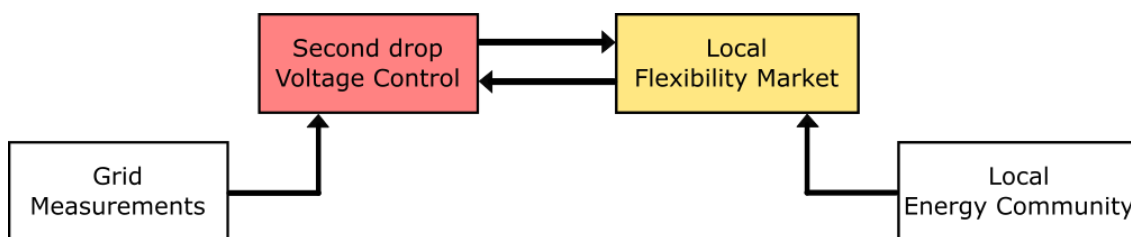


Figure 2: Voltage Control and Local Flexibility Market interface

The control strategy developed in Task 1.3 is part of the edgeFLEX architecture, described in D4.1. Therefore, Task 1.3 has focused not only on the algorithm but also on the interface with the Flexibility Market and with the rest of the platform. The control has been developed together with communication interfaces and it has been coupled with a powerflow solver to verify the algorithm in a simulated environment.

1.1 Objective of the report

The report covers both the description of the voltage control developed for Task 1.3 and the interface with the Local Flexibility Market. In addition, the control algorithm, implemented in a docker container and communicating with a simplified Flexibility Market is validated on a LV distribution grid.

1.2 Outline of the report

The report consists of four main parts. The first chapter describes the concept of the integration of the market in the voltage control. The second chapter presents the theory of the Model Predictive Control (MPC) applied to the specific voltage control case. The third chapter presents the interaction with the Flexibility Market. The last chapter presents a validation on a 40 nodes LV grid.

1.3 How to Read this Document

This report can be read as a standalone document. However, other deliverables can be helpful to get a better view of the concepts advanced in the edgeFLEX project and to have more details on the edgeFLEX platform. In particular, other deliverables closely related to this one are:

- D1.1 - Scenario description for dynamic-phasor driven voltage control for VPPs: This deliverable starts with an overview of the state of the art in terms of VPPs and of services developed within their framework. Moreover, D1.1 introduces to the reader the dynamic-phasor driven voltage control scenario.
- D1.2 - Dynamic-phasor driven voltage control concept for current VPPs in large scale deployment deliverable: This deliverable describes the voltage control algorithm that has been delivered in the first part of the project as a docker container for integration in the edgeFLEX architecture.
- D4.3 - Description of internal interfaces for control services: This deliverable in particular describes the internal interfaces which allow the edgeFLEX platform to interact with Grid Control services and also external services such as VPP optimisation and engaging with Flexibility trading platforms.

2. Integrating the Market in the control to engage the customers

In the previous implementation of the voltage control in D1.2, the Distributed Generators (DGs) and the Energy Storage Systems (ESSs) were considered directly controllable by the control algorithm. However, this can be considered valid only in the case the control acts on assets directly belonging to the DSO. In this condition, the assets are directly accessible by the DSO and the voltage control can control them to reach the optimal technical solution and counteract the undesired voltage condition.

The interaction with the customers is less straightforward, since the assets are not owned by the DSO, which also means that the DSO has no direct control over them. However, the customers, as part of an energy community, participate in the local Flexibility Market to trade their active and reactive power availability within a time window. Therefore the control algorithm can use the Flexibility Market to engage the customers of the energy community to participate in the voltage control.

The control algorithm needs to determine, based on measurements and forecast values or scheduled values, the values of active and reactive power set-points required for a future time window. Therefore, the control algorithm that has been chosen to implement this approach is the Model Predictive Control (MPC), which is described in Chapter 3. The result of the voltage control is shared with the local Flexibility Market as flexibility prevision, representing a request for active and reactive power for a time window.

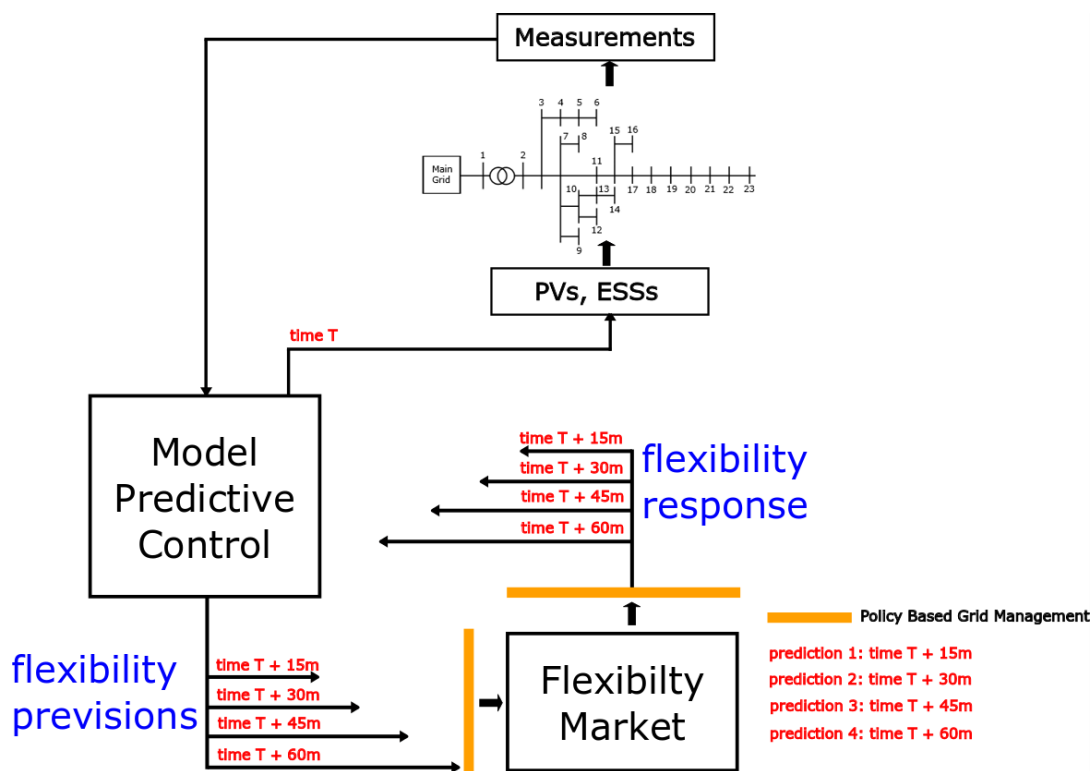


Figure 3: Interface between the MPC and Flexibility Market

The market takes the flexibility prevision as input and performs the local trading to try to satisfy the request. However, the result of the trading, defined as flexibility response, can be different from the previsions calculated by the MPC. Nevertheless, the result of the trading must be followed, given that it has been agreed among the customers in the market. Therefore, the MPC, once the flexibility response has been received, performs a second step of the control to compensate for any deviations due to the market, considering also the assets belonging to the DSO.

The coordination between the MPC and the Local Flexibility Market is described in Figure 3, where the MPC, based on the measurements (and the schedule or forecast), calculate flexibility previsions to compensate for a voltage problem with a time interval of 15 minutes. The market

based on the previsions, calculate the response and send the result back to the MPC for the additional control step. The final result is sent to the customers' and DSO's assets.

3. Model Predictive Control (MPC)

As described in Chapter 2, the Model Predictive Control has been selected as control strategy to be interfaced with the Flexibility Market. This chapter presents the control theory, the algorithm implementation whereas some preliminary results to demonstrate the MPC implementation and interface with the market are presented in Chapter 5.

3.1 Algorithm Description

The control developed for the second drop of the algorithm, is based on the Model Predictive Control (MPC), which is used to solve the optimization for future time steps within a prediction horizon N_p . To perform the optimization, the MPC makes use of measurements from the field and the results of the scheduling [1].

The general theory for the MPC requires these three components:

- A model of the system, derived from the grid topology, which is used to predict the evolution of the system in the open-loop.
- An objective function to be solved over a finite horizon. The role of the MPC is to minimize the objective function that is subject to constraints and to the system model, possible restrictions can be imposed on control inputs and state.
- A closed loop with measurements, to compensate for disturbances.

In the condition that the MPC follows the result of a scheduler, the objective function can be chosen to minimize the deviation from the operating plan and to counteract potentially predicted undesired voltage conditions [2].

The MPC minimizes the objective function for the whole prediction horizon, meaning for N_p steps with length Δ_k .

The choice of the time step length could be in general different from the time-step obtained from the forecast or from the schedule. In that condition, interpolation methods must be applied. After the calculation of the control set-points for the N_p steps, only the first control output of the obtained control sequence is transmitted to the controllable assets. After that, the optimization is executed again over a shifted horizon at time $k+1$.

To close the loop and obtain the initial condition to calculate the predicted control output, the MPC uses voltage and power measurements from the field. With this approach, the control is more robust against the forecast errors, given that any forecast error is compensated by the real measurements and on the other side, the voltage violations can be predicted and counteracted in advance while trying to track the reference.

With the proposed MPC it is possible to control the active power injection of the Energy Storage Systems (ESSs) and the active power curtailment and reactive power injections of the Photovoltaics PVs.

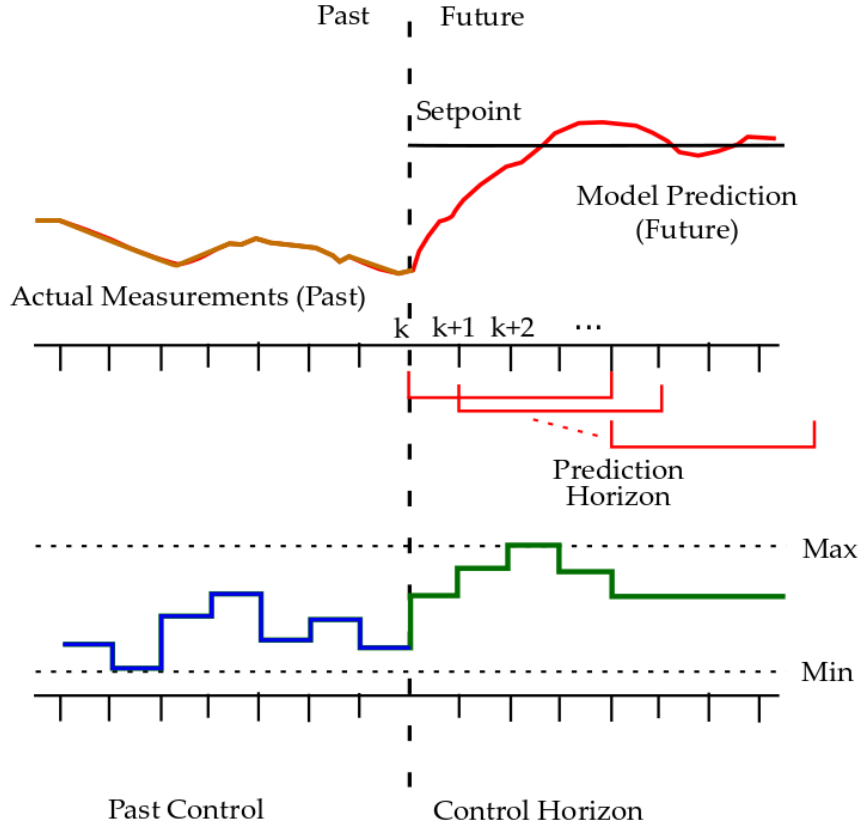


Figure 4: MPC control description

The control output, resulting from each optimization step, is provided every k . The MPC includes the active power curtailment P_{DG}^{curt} and the reactive power Q_{DG} as optimization variables together with the active power P_{ESS} . The set-points of the optimization calculated at the step $k + s$ of the prediction horizon N_p are defined as $P_{DG,k+s|k}^{curt}$, $Q_{DG,k+s|k}$, $P_{ESS,k+s|k}$, where:

- $P_{DG,k+s|k}^{curt}$ is the vector of active power set-points assigned to PVs calculated at prediction $k + s$
- $P_{ESS,k+s|k}$ is the vector of active power set-points assigned to ESSs calculated at prediction $k + s$
- $Q_{DG,k+s|k}$ is the vector of reactive power set-points assigned to PVs calculated at prediction $k + s$

The use of the subscript $x_{k+s|k}$ refers to the prediction of the generic variable x at the future time step $k + s$ given information at the time k .

The calculation of the control output is also based on the values $P_{DG,k+s-1}^{curt}$ and $P_{ESS,k+s-1}$ and $Q_{DG,k+s-1}$. If $s = 1$, the vectors represent the measurements that are collected from the sensors and devices installed in the field, whereas if $s > 1$ they represent the previous calculated control output of the prediction horizon.

Therefore, the resulting objective function $C(P_{DG,k+s|k}^{curt}, Q_{DG,k+s|k}, P_{ESS,k+s|k})$ is defined as:

$$\begin{aligned} & \sum_{s=1}^{N_p} (P_{DG,k+s|k}^{curt} - P_{DG,k+s|k}^{ref})^T W_P (P_{DG,k+s|k}^{curt} - P_{DG,k+s|k}^{ref}) \\ & + (Q_{DG,k+s|k} - Q_{DG,k+s|k}^{ref})^T W_Q (Q_{DG,k+s|k} - Q_{DG,k+s|k}^{ref}) \\ & + (P_{ESS,k+s|k} - P_{ESS,k+s|k}^{ref})^T W_{P,ESS} (P_{ESS,k+s|k} - P_{ESS,k+s|k}^{ref}) \end{aligned}$$

where:

- N_{tot} is the total number of nodes

- $P_{ESS,k+s|k} \in R^{N_{tot}}$, $P_{DG,k+s|k}^{curt} \in R^{N_{tot}}$, $Q_{DG,k+s|k} \in R^{N_{tot}}$
- $P_{ESS,k+s|k}^{ref} \in R^{N_{tot}}$, $P_{DG,k+s|k}^{ref} \in R^{N_{tot}}$, $Q_{DG,k+s|k}^{ref} \in R^{N_{tot}}$ represents the corresponding reference values
- $W_{P,ESS} \in R^{N_{tot} \times N_{tot}}$, $W_P \in R^{N_{tot} \times N_{tot}}$, $W_Q \in R^{N_{tot} \times N_{tot}}$ are positive definite weighting matrices.

To simplify the notation, in the rest of the work the term $k + s | k$ is substituted with $\cdot | k$.

The final MPC problem, adding the constraints is defined as:

$$\min C(P_{DG,k+s|k}^{curt}, Q_{DG,k+s|k}, P_{ESS,k+s|k})$$

subject to

$$\begin{aligned} & a) V_{min} < V < V_{max} \\ & b) V_{k+s} = V_{k+s-1} + R_{ESS} P_{ESS,k+s-1} + R_{DG} P_{DG,k+s-1} + X_{DG} Q_{ESS,k+s-1} \\ & c) P_{DG,k+s|k}^{min} < P_{DG,k+s|k}^{curt} < P_{DG,k+s|k}^{max} \\ & d) Q_{DG,k+s|k}^{min} < Q_{DG,k+s|k} < Q_{DG,k+s|k}^{max} \\ & e) P_{ESS,k+s|k}^{min} < P_{ESS,k+s|k} < P_{ESS,k+s|k}^{max} \end{aligned}$$

Where:

- V_{max}, V_{min} are the vectors of the minimum and maximum limits for the voltage values
- $P_{DG,k+s|k}^{max}, P_{DG,k+s|k}^{min}, Q_{DG,k+s|k}^{max}, Q_{DG,k+s|k}^{min}, P_{ESS,k+s|k}^{max}, P_{ESS,k+s|k}^{min}$ are the vectors of minimum and maximum available active and reactive power
- V_{k+s} is the vector of voltage magnitudes calculated at timestep $k + s$ of the prediction horizon as function of the vector of voltage magnitudes calculated at previous timestep and of the active power of the PVs and ESSs and of the reactive power of the PVs

In the interaction with the Flexibility Market, described in Chapter 4, the reference values will be used take the results of the trading into consideration.

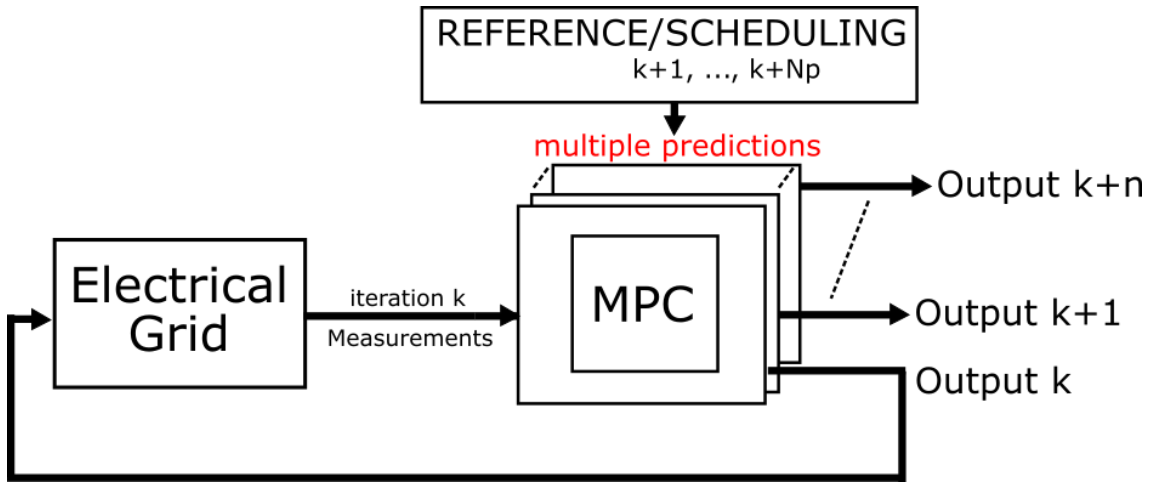


Figure 5: Block diagram of the MPC control and the interaction with the electrical grid

4. Interaction with the Flexibility Market

4.1 Calculation of the Flexibility Previsions

As described in Chapter 2, the MPC is interfaced with the Flexibility Market in order to exchange flexibility previsions and responses.

The flexibility previsions are obtained by the performing the MPC within a pre-defined time window, using the measurements and the schedule or the forecast data.

In case a voltage violation is detected, the MPC calculates the control output to compensate for it, taking into consideration assets belonging to the customers and, if present, assets belonging to the DSO. The control output to be sent to the flexibility market is defined as:

```
Flex_request = {"variable": "variable", "nodes": [n1,n2,n3,...],
"flex_request":{"prediction 1": [k1,k2,k3,...], "prediction
2": [k1,k2,k3,...], "prediction 3": [k1,k2,k3,...], ...},...}
```

Where:

- **variable:** defines which control variable is considered
nodes: defines to which nodes the request is applied
- **flex_request:** defines the previsions calculated by the MPC for each node, divided into the different prediction steps

The resulting JSON message is then transmitted to the Flexibility Market via MQTT, and then the control is set to waiting mode, until the response of the market is received.

To implement the control in the edgeFLEX architecture, the code has been organised in a git repository, as has the code presented in D1.2, to facilitate the integration in the architecture and the coordination among WP1 and WP4 activities.

The repository is composed of

- **Control_Main_MPC.py:** This is the main control file
- **Cases:** The folder contains grid data of test-grids
- **Control_strategies:** The folder contains the different functions that are called in Control_Main.py
- **Setup:** This folder contains all the files used to set the virtual environment
- **Submodules:** This folder contains some git submodules that are automatically pulled when cloning with the --recursive flag.

The algorithm requires to be instantiated in a docker container in order to be implemented in the edgeFLEX platform.

As part of the edgeFLEX architecture, the MPC needs to be interfaced with the rest of the components of the architecture. The communication protocol that has been selected to exchange data between the voltage control service and the architecture is MQTT.

4.2 Flexibility Market (basic implementation)

To test the interface between the MPC and the Flexibility Market, a dummy implementation of the Flexibility Market has been developed. This implementation does not implement a market logic, given that is out of the scope of the WP1, but simply calculates a dummy response based on the received previsions.

The calculation of the **flex_response** for each prediction j is calculated as follows:

```
Flex_response["prediction j"] = round(random*flex_request["prediction
j"])
```

The function calculates a random value between [0,1] and then it is multiplied with the request received from the MPC. This formulation, although very simple, allows the calculation of a flexibility response, mimicking the result of a real trading.

The flexibility response is sent to the MPC as an MQTT message.

4.3 MPC integrates the flexibility response

Once the MPC receives the flexibility response from the market, it integrates these values and performs a new iteration of the algorithm.

In this second iteration of the algorithm, the control considers the values obtained from the market as reference values to be tracked. Therefore the control will calculate the new set-points for the DGs that have not participated in the market and that do not belong to customers but rather to the DSO and thus are freely controllable.

4.4 Powerflow simulation

To test the MPC and the result of the integration with the Market, the powerflow service from D1.2 has been used. The service simulate the real electrical grid and it is interfaced with the MPC running in a docker container to receive set-points and send measurements.

The repository is composed of

- **runPF.py**: This is the main file for the simulation
- **Cases**: The folder contains grid data of test-grids
- **Data**: The folder contains different load and generation profiles
- **Setup**: This folder contains all the files used to set the virtual environment
- **Telegraf**: This folder contains the configuration file for Telegraf
- **Submodules**: This folder contains some git submodules that are automatically pulled when cloning with the `--recursive` flag.

4.5 Simulation Setup

Figure 6 describes the simulation setup that has been used to obtain the preliminary results for this deliverable. The MPC control service runs in a docker container and exchanges MQTT messages with the powerflow and the Flexibility Market, both also running in docker containers.

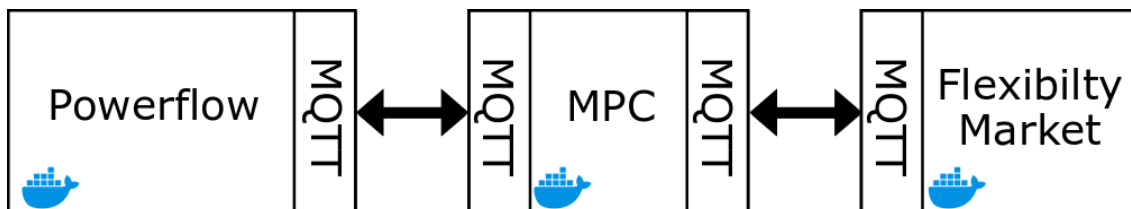


Figure 6: Simulation Setup

Figure 7 shows the schematic representation of the interaction among the Flexibility Market, the MPC and the Powerflow, showing also the interaction with the user.

First of all the user, by means of the docker-compose command, starts the containers running the three different components of the simulated system. The Flexibility Market is started in a "idle" mode, meaning that the calculation of the flexibility response is not calculated immediately but only when the user requests it.

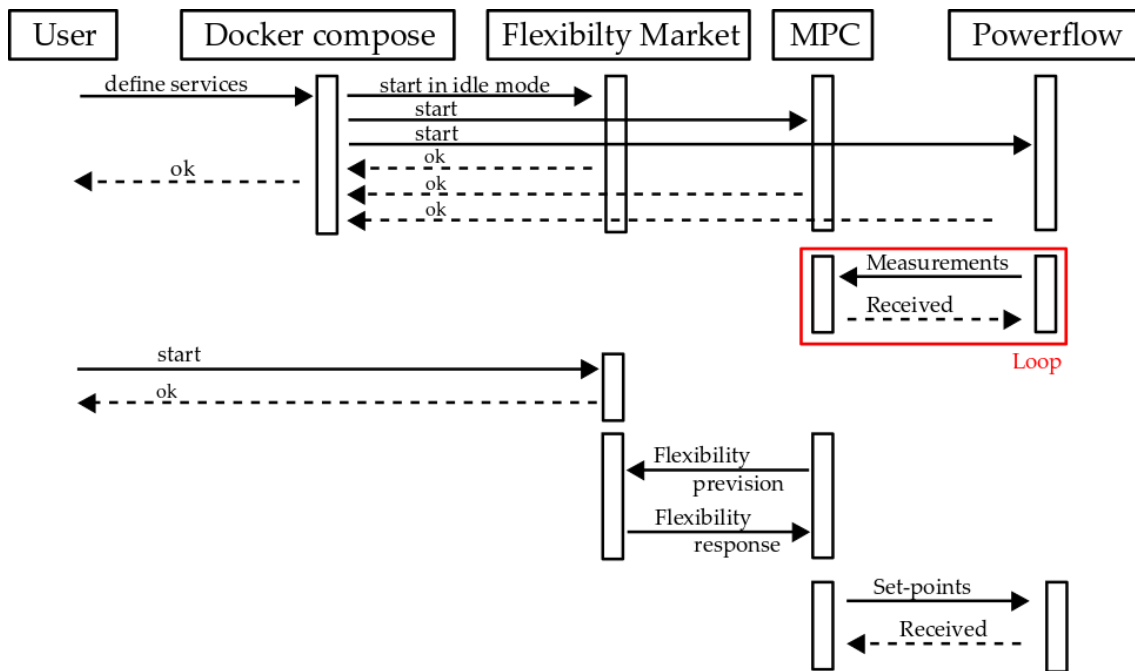


Figure 7: Interaction between Flexibility Market, MPC and Powerflow

Therefore, until the Flexibility Market is started, the MPC receives the measurements from the powerflow and, even though calculating the control set-points it does not send the results to the powerflow.

Only when the Flexibility Market is started, the MPC sends the set-points to the Flexibility Market for trading and receives back the flexibility response. After a new iteration of the MPC algorithm, the resulting set-points are sent to the powerflow.

The powerflow algorithm iterates every $t = 0.5 \text{ sec}$, whereas the MPC every $t = 2 \text{ sec}$ and they are not synchronized.

5. Preliminary results

This section shows some preliminary results of the application of the MPC control strategy in the simulation setup described in Chapter 4. The aim of this chapter is to validate the MPC control algorithm in terms of compensation for the voltage violations and interaction with the Flexibility Market.

5.1 Grid under test

The grid under test is a 40 node LV distribution grid, where on each node PV and ESS are installed and are available for the voltage control. The grid impedance values are defined in Table 1 and it is described in Figure 8.

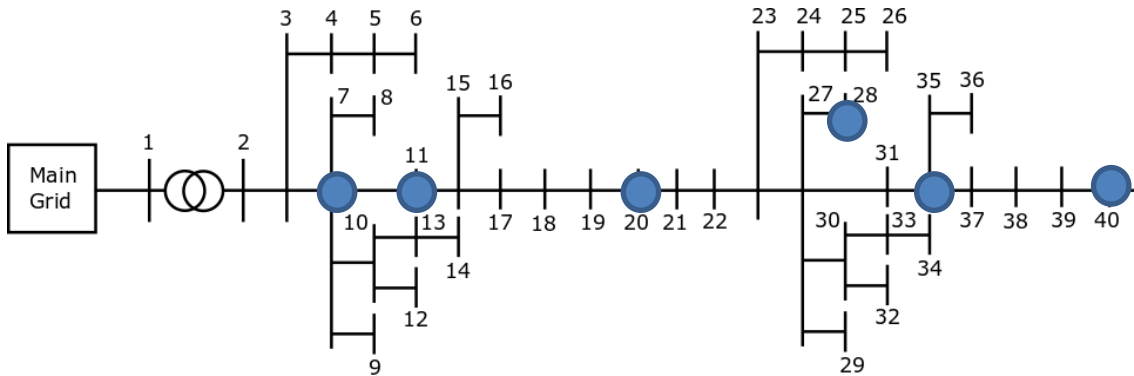


Figure 8 : 40 Nodes LV distribution grid

The nominal voltage for the grid under test is $V_{nom} = 380 V$. The limits considered in these tests are considered to be $\pm 5\%$ of the nominal value.

For the sake of simplicity, the load and generation are considered constant during the test. This assumption, however it is not unrealistic, given that the calculation of the MPC and the interaction with the Flexibility Market depends on a set of measurements taken on a certain time, therefore loads and generations could be, most likely, constant. The generators are considered all equals with constant generated power of $1.2 [p.u.]$, the loads are also equals with constant value $0.1 [p.u.]$. The DGs are 6, placed in nodes 40,35,28,20,11,7, as highlighted by the blue dots in Figure 8.

This configuration results in an initial overvoltage in the last nodes of the grid. Therefore, considering only the voltage measurements of the DGs, the resulting vector of the voltage is the following:

$$v = [1.031, 1.023, 1.025, 1.038, 1.043, 1.050, 1.053]$$

The resulting voltage profiles are also visible in Figure 9, where the uncontrolled condition is shown until iteration 20, with an overvoltage visible at node 40, where the DG number 6 is installed.

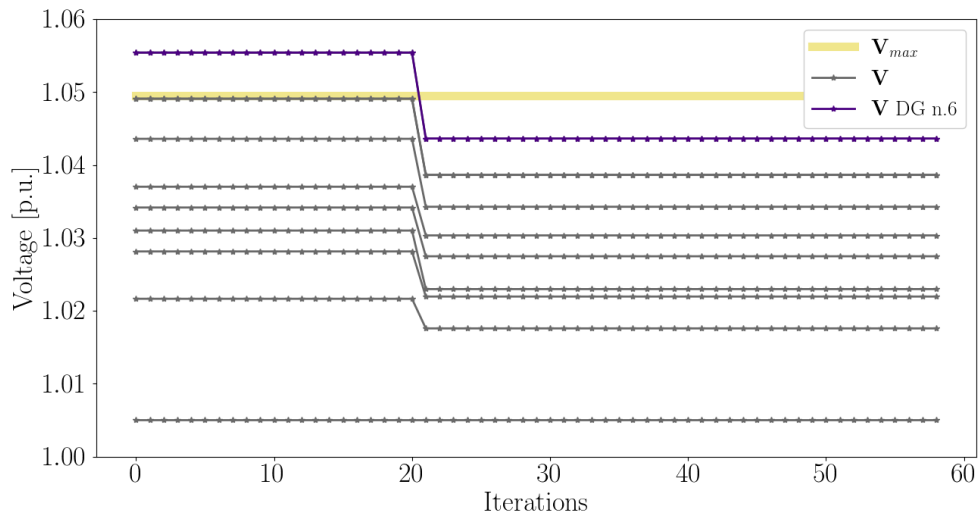


Figure 9: Voltage profiles of the nodes where DGs are installed

The MPC receives the voltage measurements and calculates the control output for the DGs as described in Figure 10 and Figure 11 for active and reactive power respectively. However, as described in Chapter 4, these values are not sent to the powerflow.

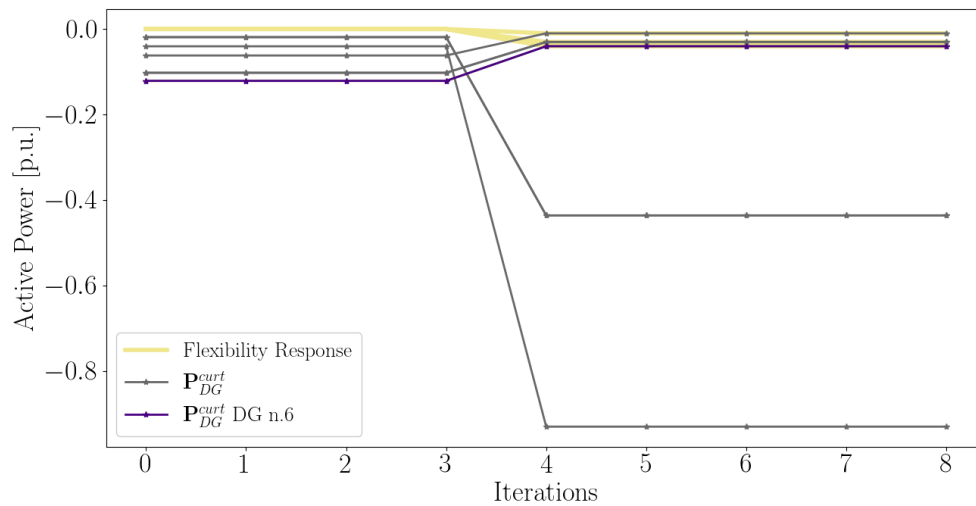


Figure 10: Active power set-points calculated by the MPC

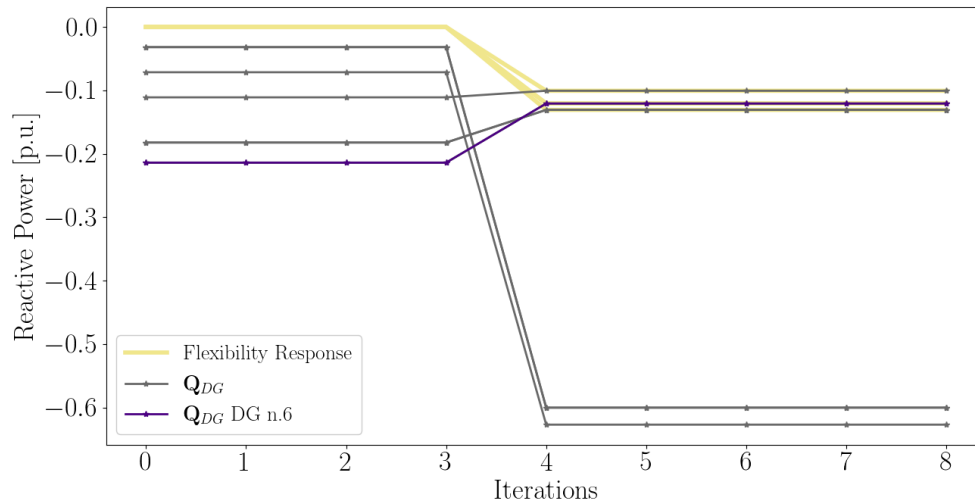


Figure 11: Reactive power set-points calculated by the MPC

Figure 12 shows the activation of the Flexibility Market by changing the value of an MPC internal variable to 1 when the MQTT message from the Market has been received. As a result, the reference values resulting from the trading, described with a yellow line in Figure 10 and Figure 11, are tracked and the values of the set-points for the other DGs are also modified. The total set-points are then sent to the powerflow and, as a result, the overvoltage problem is solved.

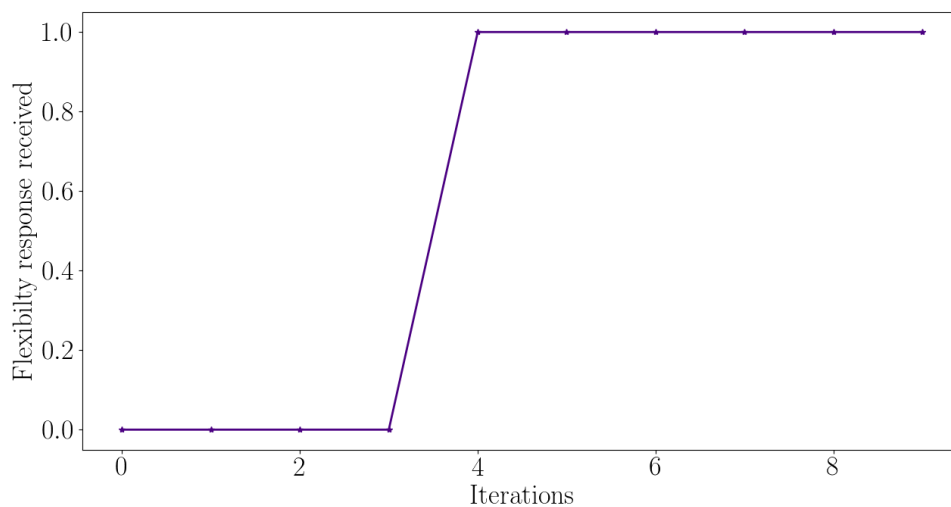


Figure 12: Activation of the flexibility

6. Conclusion

The deliverable has described the MPC algorithm for the implementation in the edgeFLEX platform and for the integration with the Flexibility Market. The algorithm, described in Chapter 3, calculates the control set-points for future time-steps based on forecast or scheduling profiles, resulting in a suitable solution for the interface with the Flexibility Market.

The optimization problem tries to minimize the quadratic function formulated with the control variables and to track, if present, reference values, while maintaining the voltage in the operational limits.

Chapter 4 has presented the interface between the MPC and the Flexibility Market in terms of communication messages and the control variables. In addition, a dummy Flexibility Market has been implemented to produce a simple flexibility response, which is integrated by the MPC for the tracking of the results of the market trading and to calculate the control set-points for the assets not participating in the market.

Chapter 4 presents some preliminary results that describe the integration of the MPC with the Flexibility Market. The MPC has been tested on a low voltage distribution grid consisting of 40 nodes. The results show that the set-points calculated by the MPC are modified once the result of the Market is received. The MPC recalculates the control output tracking the references.

For the integration of the algorithm as a service in the edgeFLEX architecture, the control has been included in a docker container. Together with the voltage control, WP1 has also provided, described in D1.2, a docker container with the powerflow service.

7. List of Tables

Table 1: 40-nodes grid data 23

8. List of Figures

Figure 1: First and Second drops of the voltage control algorithms.....	5
Figure 2: Voltage Control and Local Flexibility Market interface.....	5
Figure 3: Interface between the MPC and Flexibility Market	7
Figure 4: MPC control description	10
Figure 5: Block diagram of the MPC control and the interaction with the electrical grid.....	11
Figure 6: Simulation Setup	13
Figure 7: Interaction between Flexibility Market, MPC and Powerflow.....	14
Figure 8 : 40 Nodes LV distribution grid.....	15
Figure 9: Voltage profiles of the nodes where DGs are installed.....	16
Figure 10: Active power set-points calculated by the MPC.....	16
Figure 11: Reactive power set-points calculated by the MPC	17
Figure 12: Activation of the flexibility	17

9. References

- [1] P. D. e. a. Christofides, “ "Distributed model predictive control: A tutorial review and future research directions." *Computers & Chemical Engineering* 51 (2013): 21-41.
- [2] Y. e. a. Guo, “MPC-based coordinated voltage regulation for distribution networks with distributed generation and energy storage system.” *IEEE Transactions on Sustainable Energy* 10.4 (2018): 1731-1739.

10. List of Abbreviations

DG	Distributed Generator
ESS	Energy Storage System
JSON	JavaScript Object Notation
LV	Low Voltage
MQTT	Message Queue Telemetry Transport
MPC	Model Predictive Control

11. Appendix

11.1 Grid Data

Start Node	End Node	Resistance [p.u.]	Reactance [p.u.]	Start Node	End Node	Resistance [p.u.]	Reactance [p.u.]
1	2	0.0004	0.003172	21	22	0.00025875	0.0000975
2	3	0.00108675	0.0004095	22	23	0.000452813	0.000170625
3	4	0.000426938	0.000160875	23	24	0.000426938	0.000160875
4	5	0.00087975	0.0003315	24	25	0.00087975	0.0003315
5	6	0.0009315	0.000351	25	26	0.0009315	0.000351
3	7	0.001358438	0.000511875	23	27	0.001358438	0.000511875
7	8	3.88125E-05	0.000014625	27	28	3.88125E-05	0.000014625
7	9	0.000685688	0.000258375	27	29	0.000685688	0.000258375
7	10	0.00098325	0.0003705	27	30	0.00098325	0.0003705
7	11	0.000711563	0.000268125	27	31	0.000711563	0.000268125
10	12	0.00098325	0.0003705	30	32	0.00098325	0.0003705
10	13	0.0007245	0.000273	30	33	0.0007245	0.000273
13	14	0.000414	0.000156	33	34	0.000414	0.000156
11	15	0.000905625	0.00034125	31	35	0.000905625	0.00034125
15	16	0.000802125	0.00030225	35	36	0.000802125	0.00030225
15	17	0.000336375	0.00012675	35	37	0.000336375	0.00012675
17	18	0.000659813	0.000248625	37	38	0.000659813	0.000248625
18	19	0.000530438	0.000199875	38	39	0.000530438	0.000199875
19	20	0.000815063	0.000307125	39	40	0.000815063	0.000307125

Table 1: 40-nodes grid data