The DELTA project has received funding from the EU's Horizon 2020 research and innovation programme under grant agreement No 773960

D E L T A

Project Acronym:	**DELTA**
Project Full Title:	**Future tamper-proof Demand rEsponse framework through seLf-configured, self-opTimized and collAborative virtual distributed energy nodes**
Grant Agreement:	**773960**
Project Duration:	**36 months (01/05/2018 – 30/04/2021)**

DELIVERABLE D4.4
DELTA Aggregator Decision Support System

Work Package	WP4 – Adaptive Aggregator DELTA Supervisory Engine
Task	T4.4 – Decision Support System for the DELTA supervisory engine
Document Status:	Final
File Name:	[DELTA]_D4.4_Final
Due Date:	31.10.2020
Submission Date:	December 2020
Lead Beneficiary:	CERTH

Dissemination Level	
Public	X
Confidential, only for members of the Consortium (including the Commission Services)	

Authors List

Leading Author				
First Name		Last Name	Beneficiary	Contact e-mail
Ioannis		Koskinas	CERTH	jkosk@iti.gr
Georgios		Skaltsis	CERTH	gskaltsis@iti.gr
Co-Author(s)[1]				
#	First Name	Last Name	Beneficiary	Contact e-mail
1	Apostolos	Tsolakis	CERTH	tsolakis@iti.gr
2	Venizelos	Venizelou	UCY	venizelou.venizelos@ucy.ac.cy
3	Alexis	Frangoullides	UCY	frangoullides.alexis@ucy.ac.cy
4	Alexandre	Lucas	JRC	alexandre.lucas@ext.ec.europa.eu
5	George	Karagiannopoulos	HIT	g.karagiannopoulos@hit-innovations.com

Reviewers List

Reviewers			
First Name	Last Name	Beneficiary	Contact e-mail
Fivos	Therapontos	EAC	ptherapo@eac.com.cy
Carolina	Fernandes	KIWI	cfernandes@kiwipowered.com
Dimos	Ioannidis	CERTH	djoannid@iti.gr

Legal Disclaimer

The DELTA has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 773960. The sole responsibility for the content of this publication lies with the authors. It does not necessarily reflect the opinion of the Innovation and Networks Executive Agency (INEA) or the European Commission (EC). INEA or the EC are not responsible for any use that may be made of the information contained therein.

Copyright

[1] Special thanks to KIWI colleagues for their support and information provided for better understanding DR markets

Executive Summary

During the last years there is a growing body of research over smart grids' technology which aims to make electricity generation greener and more efficient. Demand-side management is a key part of this body, since it focuses on alternating the demand behaviour. Among other goals, efficient demand-side management facilitates a higher integration of renewable energy sources into the grid.

The DELTA project addresses the problem of designing a novel architecture in order to handle a large number of low and medium scale prosumers and ultimately flatten the demand curve. This is done by optimally participating in the electricity markets and by serving demand response requests that are made by the electric grid operator.

The document at hand provides detailed information about how electricity markets work and what the main characteristics of demand response strategies are. However, the main objective of this work is to document the architecture behind the Decision Support System of the DELTA aggregator which functions as the supervisory intelligence of the DELTA infrastructure.

To further expand the potential of the DELTA framework, some improvements are also presented in terms of forecasting services for the Net Imbalance Volume and the Day-ahead and Intra-day market prices.

The system proposed, can handle both incoming Demand Response (DR) requests from higher level stakeholders (e.g. DSO, TSO, etc.) at any given time, while also self-optimizing the Aggregator's participation to dynamic DR markets. As the markets, systems, and technologies related to DR are getting closer and closer to real-time operation, the DELTA DSS mainly focused in Day-ahead scenarios.

Table of Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

Term	Description
ARIMA	Autoregressive Integrated Moving Average
BRP	Balance Responsible Party
BSC	Balancing and Settlement Code
CR	Capacity Requirements
DA	Day Ahead
DR	Demand Response
DRM	De-Rated Margin
DSO	Distribution System Operator
GB	Gradient Boosting
ITSD	Initial Transmission System Demand
MS	Member State
NETSO	National Electricity Transmission System Operator
NIV	Net Imbalance Volume
LOLP	Loss Of Load Probability
RES	Renewable Energy Sources
RF	Random Forest
SO	System Operator
SP	Settlement Period
TSO	Transmission System Operator
XGBoost	Extreme Gradient Boosting

1. Introduction

1.1 Scope and objectives of the deliverable

This deliverable is associated with Task 4.4 of the DELTA project and provides information about the architecture of the DELTA Decision Support System. In addition, related work and methods are presented throughout the document as part of a literature review for the topic.

1.2 Structure of the deliverable

The work presented in this deliverable is structured as follows.

- **Chapter 2** introduces once more the rules and methods in EU electricity markets.
- **Chapter 3** consists of a brief literature review on the topic of Demand **-** Response strategies. The main techniques are described together with useful material about available control policies for demand-side management.
- **Chapter 4** introduces some improvements and additions following T4.3 and D4.3 in terms of forecasting services required for the DELTA DSS.
- **Chapter 5** addresses the problem and the challenge of the DELTA aggregator. The nature of instant DR requests is described and the optimal participation in electricity markets is broken down to a two-stage stochastic programming problem.
- **Chapter 6** introduces some simulated results of optimal participation in the market using artificial data.
- **Chapter 7** concludes the report.

1.3 Relation to other tasks and deliverables

The task of designing the Decision Support System (DSS) of the DELTA aggregator is related with many subcomponents of the DELTA architecture, as a supervisory engine. As it also deals with business aspects, this report is also related with WP2 results in terms of business models and outcomes. Hence, taking into account technical and business requirements from WP1 and WP2, respectively, as well as technical results from WP3, WP4, and WP5, the DELTA Aggregator’s DSS provides the supervisory engine the aims to increase the reliability and the revenues of the Aggregator’s role.

2. Energy Markets in Europe

The electricity system in Europe is going under remarkable changes during the last years. The growing role of renewable energy sources together with the fact that electric vehicles and smart energy storage systems are becoming more popular, are factors that urge the transition towards a lower-carbon and more efficient system. The energy markets play a significant role in this transition by facilitating not only more efficient generation but also more active demand-side management. Specifically, one of the core tasks of the DELTA Aggregator is to participate in the energy markets by optimally performing demand-side management. The main function of retail and wholesale markets has been previously discussed in the context of the *D2.3 "DELTA Business Models v1"*. Specifically, the potential of trading flexibility in the markets was under consideration. In this chapter, the function and structure of the main electricity markets in Europe are discussed making sure that necessary theoretical background for optimal participation is provided.

2.1 Day ahead Market

As described in [1], the day ahead market takes place on day D-1 and it concerns the bidding process through which the power agents commit to sell or buy a certain amount of energy at every hour in the day D. The energy price at each hour is not known at the time of bidding and it is only known after the market clearing process is over around noon of day D-1, when all the bids from all the generators are revealed. The output of the clearing process of the day ahead market is how much energy is to be bought or sold by each participant in every hour of day D and the corresponding price. Participation in this market is cumbersome because each participant must submit its bids one day in advance with a high degree of uncertainty, associated with important parameters such as renewable energy and market prices for day D, as described in [2].

2.2 Intra-day Market

To better handle the challenge of high degree of uncertainty, the market operator runs several sessions of the intra-day market [2]. In each session, the agents can buy or sell energy in order to adjust their acquired commitments in the day ahead market through a bidding process. These intra-day market sessions are run closer to the time of actual delivery of energy by the agents so less uncertain information is supposed to be available for the agents. The first session of the intra-day market for day D ends in the late evening of day D-1, with a time span including the entire day D.

2.3 Imbalance Market

Lastly, in every hour or quarter of the day D, a real-time balancing market is run to handle the deviations between the commitments in day ahead market and intra-day market and the actual delivery of energy in real-time. The imbalance market, also called balancing market, will determine the price of the deviation of the power agent with respect to what it was committed in the day ahead and intra-day market.

2.4 Other markets

Day ahead, intra-day and imbalance markets are of high interest for the DELTA aggregator to participate in. Yet, in Europe there are more mechanisms and corresponding markets that aim to robustify the electricity infrastructure. Such mechanisms and markets are discussed in the following paragraphs.

2.4.1 Capacity market and STOR

Depending on the available assets, a generator or generally a virtual power plant that participates in the electricity markets is able to sign contracts with the grid operator in order to provide it with the ability to request a certain amount of either increase of generation or reduction of consumption. These contracts are usually made on an annual basis and they are part of the so-called capacity market [3]. In practice, the grid operator purchases the right to request for spare capacity in cases that the grid is under stress. Unless it is an extreme emergency, the requests are done with a four-hour notice, so as the virtual power plant or the generator has enough time to prepare. In the context of the DELTA project, serving similar external requests from the grid operator is of great interest and many details about how this is dealt with are described in chapter 5.

An additional service that is provided in the UK electricity market is STOR, which stands for Short-term Operating Reserve. Despite the provision for balancing supply and demand of electricity via the aforementioned markets, unforeseen generation unavailability or actual demand being greater than forecasted might cause problems for stability of the system. STOR serves as a balancing service, where an external provider delivers standby or emergency power when requested to do so. Unlike the four-hour notice that holds in the Capacity market, STOR providers are required to generate power within a shorter period of time, depending on the contract with the national grid.

2.4.2 Frequency response services market

The grid operator in each country in Europe is obliged to control the system frequency at 50 Hz, plus or minus approximately 1%, depending on the grid regulations of each country. This is reassured by the operator by making contracts with providers who can meet some technical requirements. Such provision might come from generators connected to the transmission and distribution network, storage providers or aggregated demand side response [4]. In practice, there are several mechanisms under different labels that serve the same frequency control goal. Specifically, a frequency response service provided continuously is characterized as dynamic, whereas a service which is triggered only at a defined frequency deviation is called static. Moreover, time response is an additional factor that may characterize a label of a certain frequency response service.

3. Demand Response Strategies

Research and development over smart grids have attracted significant attention in the last few years. The growing participation of renewable energy sources, such as wind turbines and photovoltaic parks, was a key-element for reducing the dependency of the electricity grid on fossil fuels and nuclear power. However, due to the volatile nature of renewable energy sources, more and more attention is focused on how to efficiently manage the demand-side, so as to maximize the potential integration of renewables. As a matter of fact, demand response is one of the most cost-effective and reliable ways for smoothing the demand curve, in case that the system is under stress [5]. In the following paragraphs different schemes and approaches of demand response strategies are reviewed and discussed.

3.1 Classification of DR programs

Demand-response (DR) is one of the main activities for demand-side management and it can be described as the set of approaches which seek to alternate the demand behaviour of the consumers, in time and volume, so as to maximize the incorporation of renewable energy sources (RES), as described in [5]. Generally, consumers can be large-scale, medium-scale and even low-scale prosumers. Naturally, large-scale consumers such as industrial clients are more easily integrated in demand-response programs, given that few demanding loads are easier to be controlled than many of low consumption [8]. However, recently there is a growing body of research performed regarding the incorporation of residential consumers and buildings in demand-response programs [7].

3.1.1 Control Mechanism

Demand-response schemes can be first classified with respect to the control mechanism that is considered, which can be either centralized or distributed. In the centralized mode final users communicate directly with the central intelligence unit, which seeks to optimally handle the available assets. On the other hand, in the distributed mode there is interaction between final users and related information is provided to the central agent [9].

Generally, a central agent who aggregates prosumers and seeks to employ a DR strategy can be called a Virtual Power Plant (VPP) [6]. Apart from the flow of information, DR strategies can be classified with respect to the control policy that is used. Control policies are usually divided in direct and indirect, as deployed in [7]. Briefly, direct control policies refer to issuing specific commands to controllable loads, whereas indirect control policies refer to issuing signals to the final users which might or might not alternate their operation. This topic is of high importance for the DELTA aggregator, since serving internal DRs is one of its main tasks and for that we discuss it in more detail in the following subsections.

3.1.2 Direct Control

Decisions in direct control approaches are made by an external controller that serves as the intelligent agent and has access to the status of the load under control. In the DELTA approach the central intelligence agent, that is the aggregator, has access to information about aggregated consumption, however no access to information about the status of individual loads. Thus, specific decisions are taken in two steps, one for each layer of the aggregator and the DVNs.

In the general case, direct control policies can be categorized with respect to the type of information that is exchanged between the final user’s interface and the central controller [9]. According to [12], these types are:

A. **Deferred operation**: The consumption of production of a certain amount of energy is shifted in time, as presented in Figure 1.A. The amount of power consumed or produced remains the same as well as the duration of the operation. The signal that is received by the final user's interface is of type Δt, which represents the required delay of an operation.

B. **Delta operation**: The amount of energy consumed or produced by the final user's interface is decreased or increased by an offset ΔP, called power difference, see Figure 1.B. Decrease in consumption might result in increase in duration of the operation, for example in operation of thermal loads like a heat-pump.

C. **Scheduled operation:** The central agent provides the final user with an operation schedule s , consisting of time series of power set points and time stamps, where s={(t_i,P_i)}, i in N, as shown in Figure 1.C.

D. **Direct power control:** In this case the central agent provides the final user with a power set point, P, as illustrated in Figure 1.D.

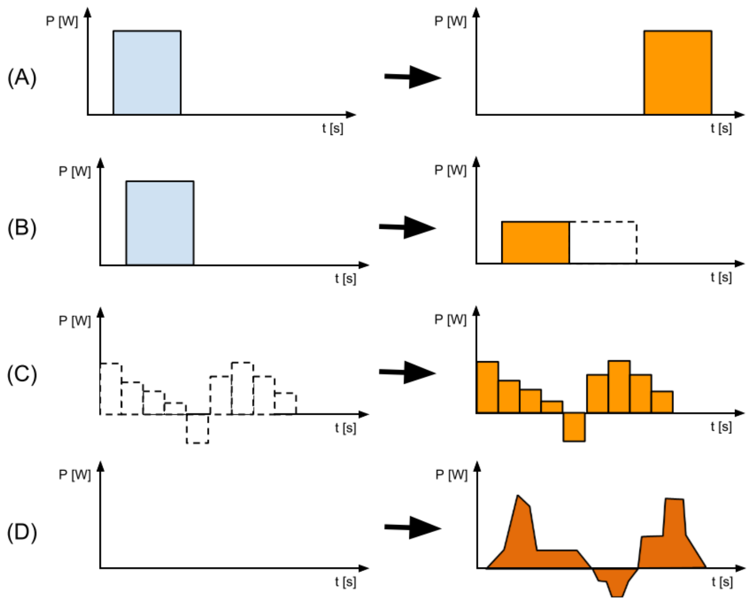


Figure 1. Direct Control Policies. A: Deferred Operation, B: Delta Operation, C: Scheduled operation, and D: Direct Power Control.

3.1.3 Indirect Control

'Indirectness of the relationship between control objective and actual outcome' and 'non-deterministic behaviour due to the fact that final decisions are taken locally and independently by final users', are the two main characteristics of indirect control policies [7]. In the general case, the final user accepts some control signals from the central agent but it is not obliged to either react to the signal or send any feedback.

Due to their nature, indirect control policies imply scalability for the system, which means that the system remains effective when there is a significant increase in number of resources or users [10]. In contrast to direct control policies where different and individual signals should be sent to each final user, in indirect control a single control signal can be received by any number of consumers. According to [11], such policies can be classified to control with indirect functional variables and indirect control via price signals.

The most commonly found in the literature is indirect control via price signals. This approach is often met as incentive-based indirect control and in the case that incentives are financial, or they can be interpreted as such, the same approach is followed. In the general case, the central agent issues energy prices or financial incentives to the final users in order to alternate their operation. Usually, the signal is a schedule *s* of future prices-incentives, consisting of time series of prices and time stamps, where:

$$s = \{(t_i,p_i)\}, i \text{ in } N.$$

In the context of the DELTA project, the control mechanism that is used is a combination of both centralized and distributed modes. Namely, the aggregator layer is responsible for high level tasks, such as participation in the markets and serving external DR requests, yet the DVNs layer is responsible for implementing the decisions made by the aggregator. Similarly, both direct and indirect control policies are necessary to be used depending on what kind of action is to be served and of course depending on the type of the final user. More details about the implementation of the Decision Support System of the DELTA aggregator are provided in chapter 5.

4. Day-Ahead, Intra-day, and NIV Forecasting improvements

Following the activities and results presented in D4.3, and given certain progress, improvements and new ideas explored through WP4 activities and close collaboration with various stakeholders, additional or improved forecasting engines have been introduced, and are documented in this chapter, complementary to D4.3.

4.1 Day Ahead - Intraday Price Forecasting

4.1.1 *State-of-the-Art*

Energy price forecasting plays an important role for the planning, the bidding strategies and consequently the risk management of market participants. Thus, a huge effort is made to predict the price of energy more accurately, especially on the part of energy companies. However predicting the energy price is a challenging task, as the high volatility of the price results in inconstant mean, variance and significant outliers. The main reason that causes the price volatility is the uncertainty and the large deviations of the solar and wind forecasts. This volatility increases as the integration of intermittent sources of electric power generation continues to rise. Finally, another important factor that affects the price forecasting are the fuel prices such as oil and natural gas prices. Many attempts have been made in order to tackle these challenges. Generally, the main models that were utilized can be classified into two categories, namely time series models and machine learning models. The most commonly used time series model that is applied in electricity price forecasting is the autoregressive model and more specifically its variants, such as autoregressive integrated moving average (ARIMA), autoregressive and moving average (ARMA) and generalized autoregressive conditional heteroscedasticity (GARCH)[2] [16][17]. Besides, [18] propose a new time series model called autoregressive-GARCH. Last but not least, some time series models have been combined with other models [19][20]. Regarding the machine learning models artificial networks are widely used, due to their ability to learn and represent accurately complex and non-linear patterns. In [13] a day ahead price forecasting is conducted by utilizing neural networks. Additionally, there is an interesting comparison between many different models that are using different numbers and types of features input. More particular, the model with the best performance seems to be the one that is using historical data of natural gas and predicted RES values. Ensemble methods were also utilized by combining different machine learning models [21] and achieving a higher accuracy at the end. Finally in the recent years, there have been many efforts to integrate and utilize features extracted from the connected electricity markets [14][15], in order to improve the predictive performance of the day ahead price forecasting.

As it was mentioned above due the uncertain nature of the renewable energy sources (RES) there is a high possibility for an inaccurate forecast resulting in high real time prices. In order to avoid the risk of inaccurate forecasts and to follow some possible updates in the available conventional power plants, day-ahead markets have been complemented by intraday markets. Though, variable generators and load serving entities prefer to participate in intraday markets and balance their positions by trading energy closer to the operating hour. As it becomes clear the only difference with day-ahead market lies in the fact of their closeness to the real delivery time. The intraday markets that are organized by power exchanges usually take the forms either of auctions or continuous trading. Purchasing and selling of electricity is allowed throughout the whole day, up to a few minutes before the physical delivery. The objective function of the intraday auctions is to minimize the total adjustment cost, with respect to the deviations from day-ahead market results. For all the aforementioned reasons above, intra-day electricity price forecasting research has received increasing attention. In [22] there is a proposed forecasting strategy in order to detect and capture the spikes in electricity price forecasting, which are unlikely to be found by the day electricity price forecast. Additionally, [23] reports about the economic benefits having precise intraday price forecasts.

---

[2] https://www.sciencedirect.com/topics/engineering/heteroscedasticity

The electricity price forecasting tool developed within the DELTA project, takes into consideration both day ahead and intraday markets and more specifically the energy market of the United Kingdom, but it can be applied to other energy markets as well. As a result, two forecasting tools were developed, namely a day-ahead price forecasting tool and a complementary tool corresponding to the intraday price forecasting. Day-ahead is executed every day and more particular every midnight, unlike the intraday tool that is executed in half-hour time intervals. As it was mentioned before, the only difference between the tools lies in the fact of their time execution. The goal of executing every half-hour the intraday forecasting tool is to detect and capture the spikes that often occur in the electricity prices. Below a flow diagram describing the methodology for the development of the electricity price forecasting tool is illustrated.

4.1.2 Methodology

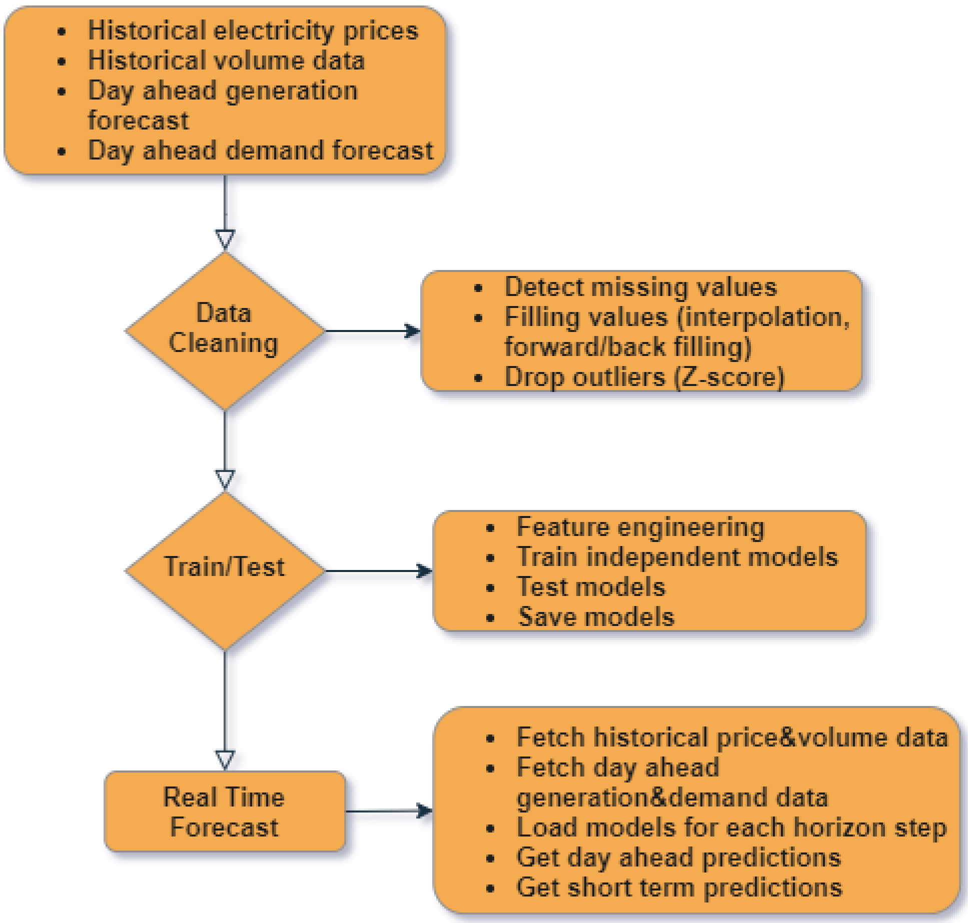


Figure 2. Day-ahead and Intra-day forecasting methodology flow.

4.1.2.1 Data Cleaning:

Data cleaning is one of the most important and time consuming aspects during time series forecasting. In many cases during the data collection, data will be either not recorded or will be stored at the wrong value. The data cleaning module is responsible for detecting such values and managing them accordingly by replacing, modifying or dropping them. The main goal is to construct time series out of the data provided by the local database. Because the prediction models use records collected for a whole day, missing data is either omitted or supplemented in case the number of the missing values is lower than a certain percent for every day respectively. Finally, values that are considered outliers are removed using the z-score method. Z-score describes the position of a record in terms of the distance from the mean, when measured in standard deviation units.

4.1.2.2 Train/Test process:

The feature engineering and development of the models is the next step after the data cleaning module. In general, predicting multi-step time series forecasting problems can be achieved in two ways, namely with direct or with recursive multi-step forecast strategy. The main difference of direct and recursive multi-step lies in the fact that the recursive is utilizing the forecast for one step ahead as a new input feature for the next forecast, unlike with the direct strategy where for each horizon step an independent model is trained respectively. Below there are the equations that represent both strategies:

Recursive multi-step strategy:

$$prediction(t+1) = model(value(t-1), value(t-2), \cdots, value(t-n))$$

$$prediction(t+2) = model(prediction(t+1), value(t-1), \cdots, value(t-n))$$

$$\ldots$$

$$\mathrm{prediction(t+n)} \ = \ \mathrm{model(prediction(t+n-1), prediction(t+n-2), \ldots, value(t-1))}$$

Direct multi-step strategy:

$$\mathrm{prediction(t+1)} = \ \mathrm{model1\big(value(t-1), value(t-2), \ldots, value(t-n)\big)}$$

$$\mathrm{prediction(t+2)} \ = \ \mathrm{model2(value(t-1), value(t-2), \ldots, value(t-n))}$$

$$\ldots$$

$$\mathrm{prediction(t+n)} \ = \ \mathrm{model}n\mathrm{(value(t-1), value(t-2), \ldots, value(t-n))}$$

The direct multi-step strategy was adopted for the price forecasting tool, as it becomes clear from the above equations that in the recursive strategy the final forecasting contains the accumulated error from the previous forecasts. The only drawback for the direct strategy is that the development of many models adds a computational effort, especially when the number of the steps increase.

After drawing up the strategy for the forecasting models, the selection of the input features follows. A table containing the input features is provided below:

Table 1. Input parameters for the day-ahead/intra-day price forecasting engine.

Prediction Type	Execution Time	Time Resolution	Input Features	Number of GBTs	outputs
Day-ahead	Every midnight	30 minutes	• 48 values of historical price data • 48 values of historical volume data • 48 values of forecasted day ahead generation data	48	48 price forecasting values

			• 48 values of forecasted day ahead demand data • 2 values of month of forecasted step (sinus, cosinus) • 2 values of weekday of forecasted step (sinus, cosinus) • 2 values of time of forecasted step (sinus, cosinus)		
Intraday	Every 30 minutes	30 minutes	• 48 values of historical price data • 48 values of historical volume data • 48 values of forecasted day ahead generation data • 48 values of forecasted day ahead demand data • 2 values of month of forecasted step (sinus, cosinus) • 2 values of weekday of forecasted step (sinus, cosinus) • 2 values of time of forecasted step (sinus, cosinus)	48	adaptive number of price forecasting values

As it was mentioned in the previous section the intraday is utilizing the same model as the day ahead with the only difference being that the day ahead is executed only at midnight using only the previous days data, whereas the intraday is executed every thirty minutes based on the most recent historical values. The features listed in the table above were based on the literature and on experiments as well. Representative figures of day ahead demand, aggregated generation and historical volume data for the period of one week are depicted respectively:

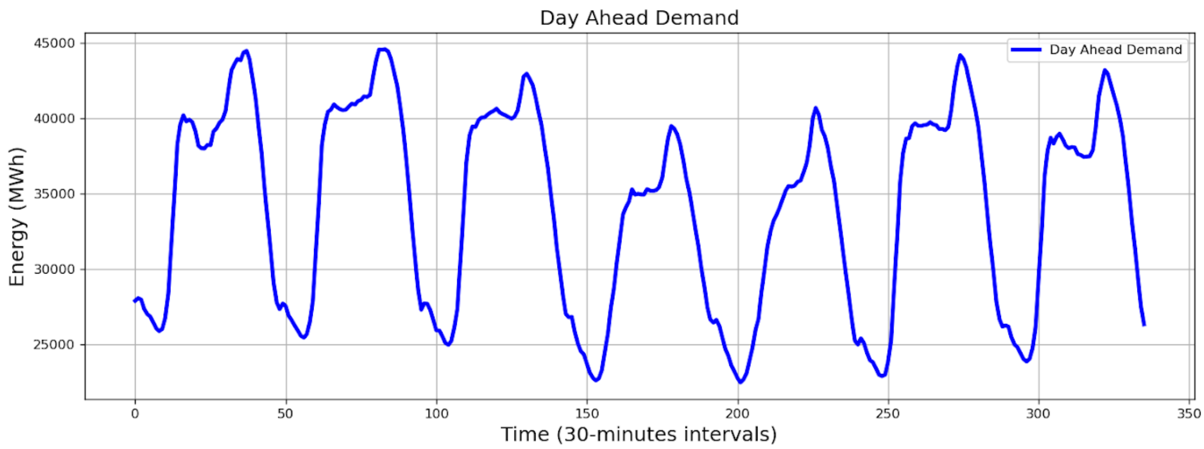


Figure 3. National day-ahead load forecast for one week

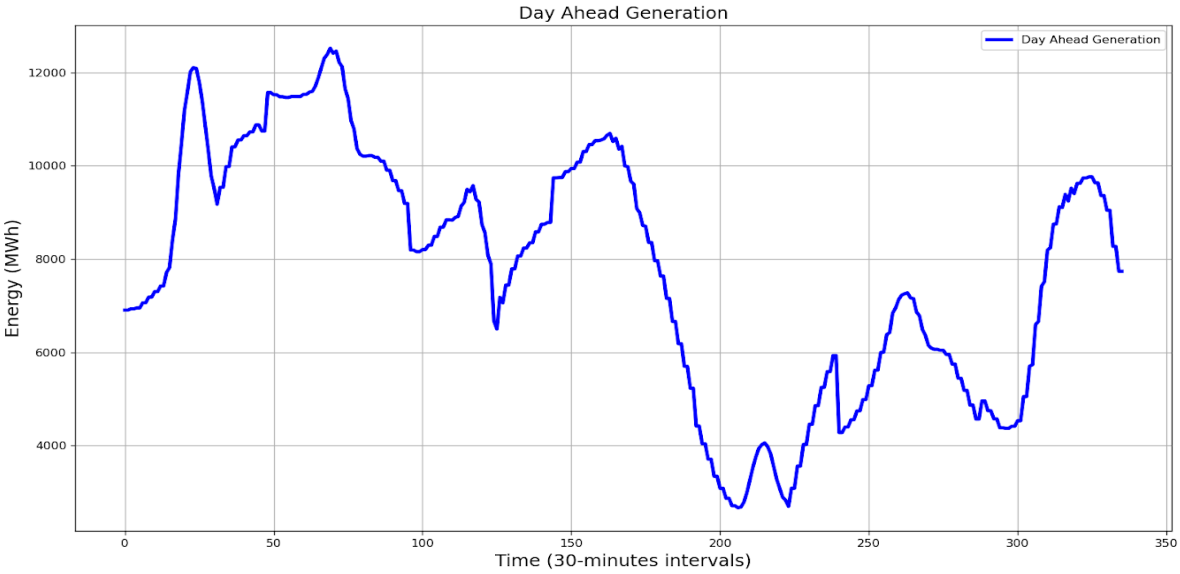


Figure 4. Aggregated day-ahead generation forecast for one week

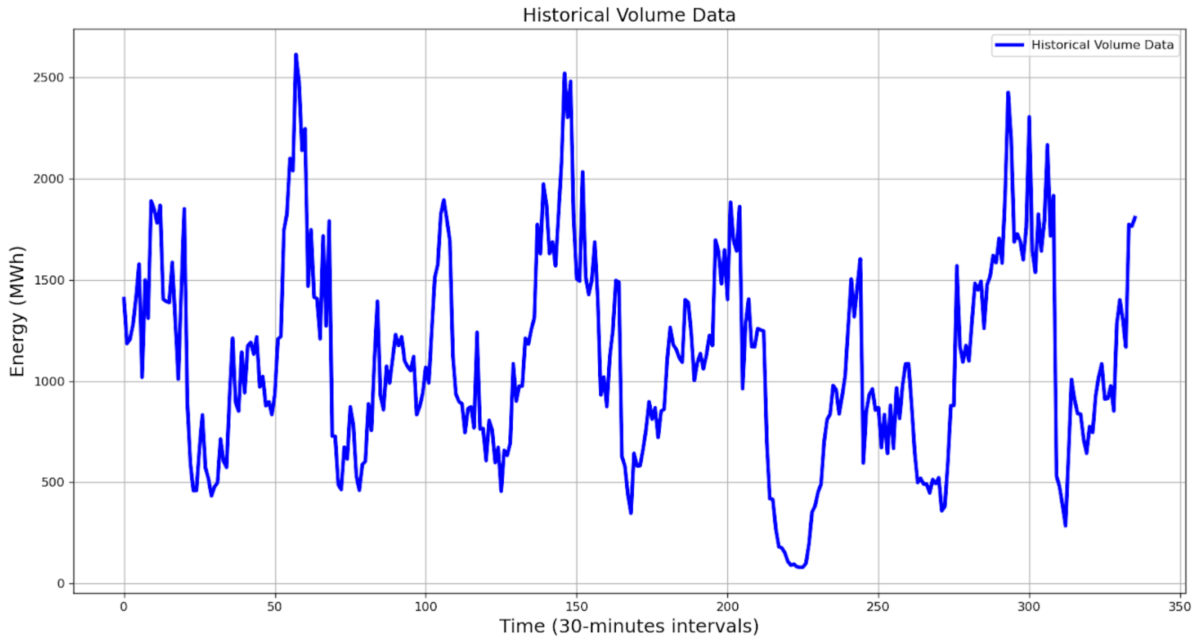


Figure 5. Historical volume data for one week

For the training, the GBT (Gradient Boosting Tree) was utilized for each model of each time horizon respectively. Gradient boosting is a general term referring to a class of ensemble machine learning algorithms that can be used for predictive problems. The construction of the ensembles consists of decision tree models. At each moment of the training a new tree is added and fit to correct the errors made by prior models. This type of training is known as boosting. More specifically a variant of GBT was utilized, namely the LightGBM, short for Light Gradient Boosting Machine. LightGBM [24] is a relatively new algorithm and is becoming more and more popular, due to its superior performance in terms of speed and accuracy, especially in predictive problems. While other algorithm trees grow levelwise, LightGBM is growing vertically, meaning that LightGBM grows leaf-wise. The leaf with max delta loss is chosen, resulting in a better accuracy. Finally, LightGBM can handle large datasets and takes lower memory, increasing the execution time of the training. Below a table with the comparison results between training models are provided. The evaluation of the models is conducted in terms of accuracy based on regression metrics and execution speed.

Table 2. Summary results of the deployment of different ML models

Forecasting Methods	**SMAPE(%)**	**MAPE(%)**	**RMSE(£)**	**Execution Time**
Light Gradient Boosting Machine (LightGBM)	6.44	1.59	7.14	122.58
Extreme Gradient Boosting (XGBoost)	6.84	1.76	7.5	190.84
Multilayer Perceptron (MLP)	7.38	1.93	7.59	390.97

The training of the algorithms was carried out with three months of historical data, including all the features described above. The last week of the historical data was used for the evaluation and the testing of the accuracy. Three different models were implemented and LightGBM surpassed the other two models, both in accuracy and execution time respectively. XGBoost and MLP were selected as alternatives for the training of the models, because these specific algorithms are widely used for predictive problems as well. For the evaluation of the results three indicative regression metrics were used, namely root mean squared error (RMSE), symmetrical absolute percentage error (SMAPE) and weighted root squared error (WRSE). RMSE and MAE are classic error metrics for regression problems, so there is no need to elaborate further on their functionalities. WRSE is created in order to demonstrate the relative error in terms of magnitude of the evaluated price value and it is described by the following equation:

$$\mathrm{WRSE} = \frac{\left\{\sum_{i=1}^{M}\sqrt{\frac{(\hat{y}_i-y_i)^2}{y_i}}\right\}^2}{M\,\sum_{i=1}^{M} y_i}\,100\%\text{ , where}$$

- $\hat{y}_i$: forecasted price output at every time slot
- y_i: actual price output power at every time slot
- M: number of forecasted steps ahead

A figure with the comparison between the actual and forecasted prices for the period of one week having as inputs the aforementioned features is illustrated below:

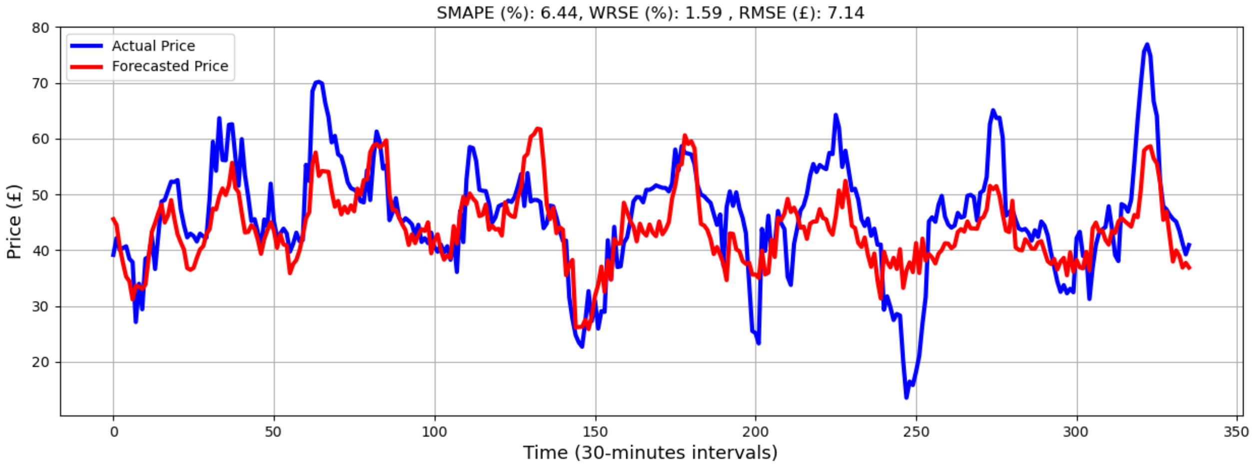


Figure 6.Price forecasting results for one week

4.1.2.3 Real Time Forecast:

The last step of the forecasting tool is the day-ahead and intra-day price forecasting. During the real time prediction the pre-trained model is loaded. The day ahead forecast is executed every midnight at 12:00 a.m. and the intraday forecast is updated every forecast step and adjusts the predictions to the latest changes of price value by using as input features the most recent historical data (historical price data, historical volume data etc.). An example of a real time comparison between day-ahead and intraday price forecasting is given below:

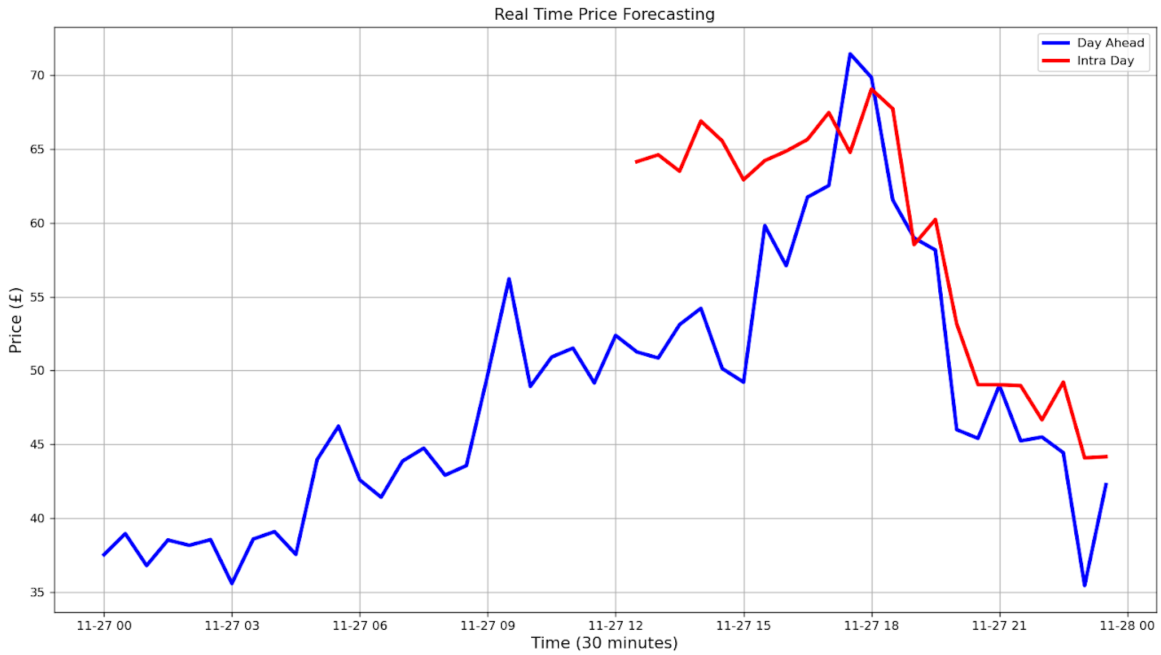


Figure 7. Day-ahead and intraday real time forecast for a certain day

In the figure above the real time price forecast of day-ahead and intraday for a certain day is depicted. The blue line corresponds to the day-ahead price forecasting, while the red line to the intraday. The intraday price forecast presents a discontinuity as the intraday forecast was executed only once in the middle of the day only for experimental reasons in order to show the functionality of the specific module.

4.2 Net Imbalance Volume Forecasting

First attempts of Net Imbalance Volume (NIV) Forecasting were presented in D4.3. In general NIV, is a highly correlated feature with the Price of the Imbalance Market. Hence, an accurate prediction of this metric could potentially lead to a more accurate Imbalance Price Forecasting. In D4.3, the most effective designed model had Mean Squared Error (MSE): 47699 and RMSE 218. The following tables display the results of the latest attempts.

Adding LTP (Trend Analysis), XGBoost

MSE	RMSE	SMAPE	MAE
48004	219	102	166

Model accuracy ~ 0.5225

LSTM Approach

MSE	RMSE	SMAPE	MAE
50723	225	185	169

Model accuracy ~ 0.5225

Adding Weather forecast, XGBoost

MSE	RMSE	SMAPE	MAE
47699	218	102	165

Model accuracy ~ 0.5255

Adding 2018 NIV parameters, XGBoost

MSE	RMSE	SMAPE	MAE
48992	221	96	168

Model accuracy ~ 0.6056

Figure 8. Error Metrics of NIV models in D4.3

As a sequel to the previous efforts, this section presents an endeavour to reduce the error metric through a data enhancement and model design perspective. Adding more complexity to the designed models, as well as, the pursuit of data that can potentially affect the Imbalance Volume. Therefore, these topics constitute the pillars of this research. In terms of the model design, this section presents two neural networks architectures:

- Long Short-Term Memory (LSTM)
- Convolutional Neural Network (CNN).

In a short analysis of our data, the figure below displays a violin plot for each month of the time period between 2017-2020. It is discernible that the Network anticipates more extreme Positive Imbalances in 2020, while the Negative Imbalances have been reduced.

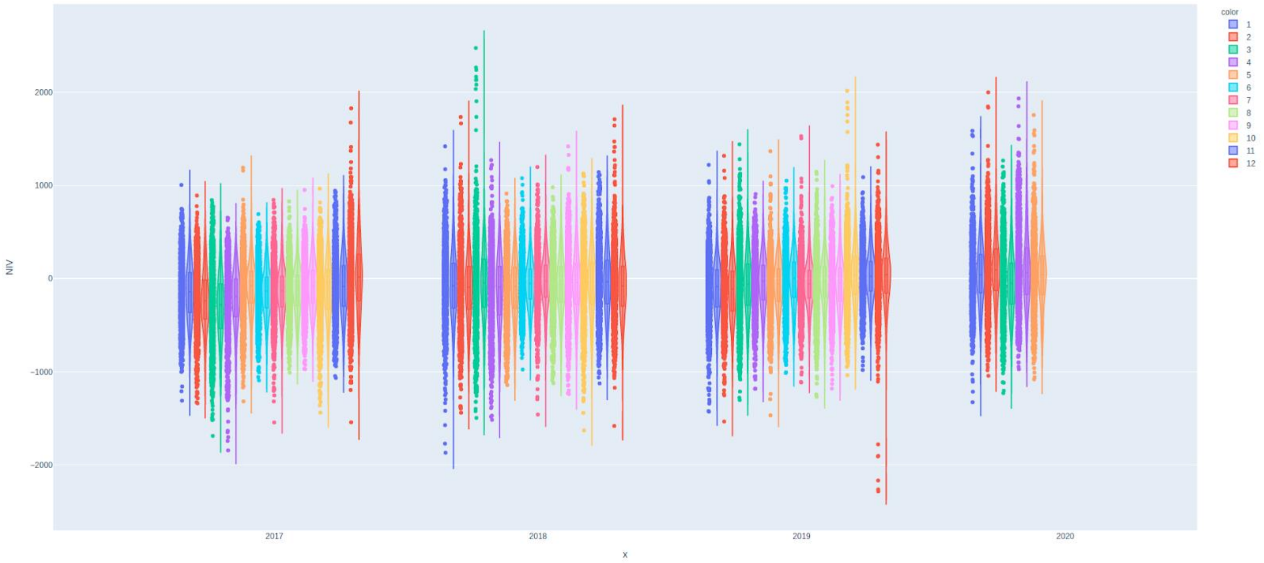
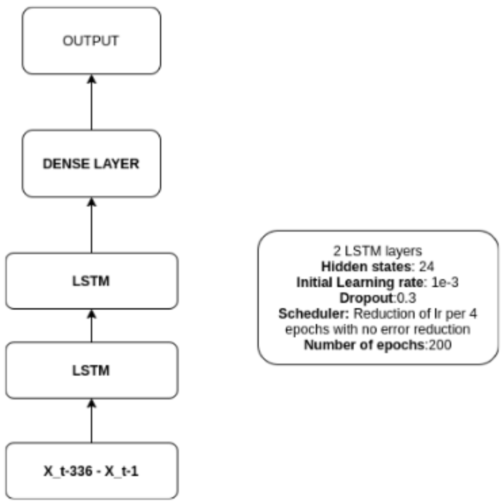


Figure 9. Violin Pilot Plot for each month of the Year.

The initial model design approach focused on the Forecasting of the Imbalance Volume through a more complex LSTM model in conjunction with a Dense Network. In this approach, LSTM acts as a feature engineering model of two layers that takes as input, 336 previous values that reflect one week period and produce a 24 length output. This vector, consequently, is given as input to the Dense Network that is responsible for the predicted Value. The following figure displays the general architecture, some technical parameters' values that have been configured towards optimal efficiency and the final Results.



Final Results

MSE	RMSE	**SMAPE**	MAE
42870	207	**93**	130

Figure 10. Network architecture, parameters and results.

Dropout technique has been applied in LSTM layers in order to avoid overfitting during the training process, while a controlled reduction in the learning rate per four epochs, ensured a smooth reduction of the error towards the exploration of the global minimum.

As far as CNN is concerned, smoothing techniques are utilized before the training of the model as a pre-processing step based on the fact that in time series analysis, these techniques affect the model's training phase positively. At the same time, in a convolutional network the first two layers - convolutional layer and pooling layer- perform smoothing and the rest of the layers use the smoothed raw data for prediction. Thus, a 1-d convolutional neural network with a fixed length window architecture has been applied and the results are depicted in Table 3.

Taking into account the outcome of the CNN model compared to the LSTM approach, the second seems to have superior performance. Hence, the research focused on the development of the aforementioned LSTM model. Additionally, a further investigation in data enhancement through an expansion of our dataset took place, inserting training data from the time period 2015-2017. Except from the extension of our dataset with regard to the time range, the feature selection research showed that energy markets prices and weather data contribute towards minimizing the forecasting error, thus they were included in the training data. More specifically, forecasted Wind offshore generated energy seems to have the highest impact in the model, while forecasted generated solar energy has the lowest one. In order to prove this assumption, Figure X presents the correlation of the weather and price features with the NIV values. Hence, the collection of previously mentioned values and feeding those into the LSTM model led to much better results.

Table 3. LSTM and CNN Results

MODEL	MSE	RMSE	MAE
CNN	44672	210	161
LSTM	40323	200.8	152.398

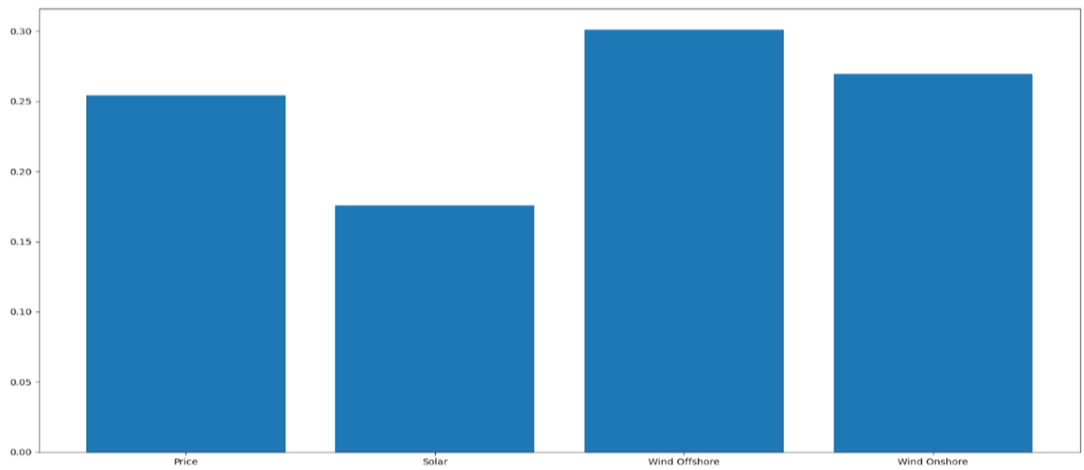


Figure 11. Features Correlation with the NIV

5. DELTA DSS Implementation

In the context of the DELTA project, the aggregator functions as a Virtual Power Plant (VPP), that is a network of decentralized low and medium scale electricity producers, such as wind farms and photovoltaic parks, as well as flexible power consumers and storage systems. The final users do not directly communicate with the aggregator. Instead, they are grouped in semi-independent clusters called DELTA Virtual Nodes (DVNs) which function as a layer in between the central agent and the final users and a detailed description about their structure is provided in Deliverable D3.2.

In this chapter we describe our proposed approach for the implementation of the Decision Support System of the DELTA aggregator. The aggregator is mainly responsible for two fundamental tasks. First, is to serve external DR requests from the grid operator. Details about this task are provided in the next section. The second and most cumbersome task of the aggregator is the optimal participation in the electricity markets. In the following sections we further deploy the details of the addressed problem and we propose a multi-stage stochastic optimization approach in order to optimize the decision-making procedure and maximize the aggregator's profits. Additionally, we describe in detail a mechanism for forecasting and scenario generation which is necessary for the optimization procedure. Finally, we present some first results in the section of simulated experiments.

5.1 Instant DR Service (IDS)

DR signals can be generated for multiple reasons and from several Sources. Either the objective is to develop, as an aggregator, a profitable DR strategy or to service an upper layer's demand, aggregator's responsibility is to manage all the available assets in the most efficient way that will lead to the fulfilment of its mission, to deliver a specific demand. One of the time constraints that concerns the IDS and the conformance to current DR policy is the fact that the assets/customers that manage to deliver a DR signal, should be noticed five hours earlier before the starting time of the DR.

This subsection concerns the instant DR service of signals that are generated from DSO/TSO or the GSSE component in order to preserve the network's grid stability. This type of signals contain information about:

- the amount of the demand
- the direction of the demand
- and the time period that needs to be serviced

As it is presented in D4.2, Energy Portfolio and Segmentation engine undertakes the Segmentation task of the total energy portfolio to individual Energy assets (DVNs). SPEB module as a subcomponent of DSS engine is responsible to select the participant DVNs to the corresponding DR according to reliability and fairness indicators. The final phase of the DR completion necessitates the distribution of the demand amount to the corresponding DVNs participants. The following figure displays the general architecture of the aggregator's layer.

In terms of the communication mechanism between all these components, the incorporation of CIM and jsonLD ontology to the DELTA ecosystem provides interoperability, security and compliance to the OpenADR communication model. Furthermore, all interactions and exchange of messages are recorded in the Blockchain system for more transparency and security reasons. Additional information about Secure Information exchange are described in D5.1 and D5.2.

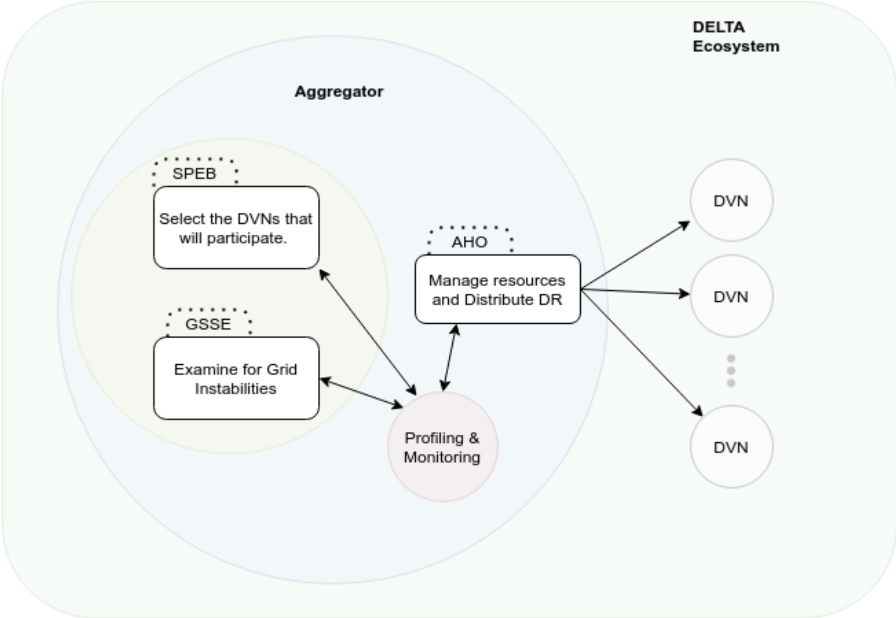


Figure 12. Workflow of the Aggregator's ecosystem

IDS addresses this problem sharing the available assets' flexibility according to the reliability metric. The following equation describes the logic of this sharing:

$$AF = \sum_{s=1}^{i+nP} (AmountFlexibility_{DVN} \cdot Reliability_{DVN})$$

where:
$AmountFlexibility_{DVN}$: Total available flexibility of the corresponding DVN
$Reliability_{DVN}$: Reliability metric of the corresponding DVN
nP: The number of DVNs that will participate in a specific DR according to the SPEB results.
AF: Represents the Aggregated flexibility of the participant DVNs in conjunction with their reliability metric.

For each DVN estimate the ratio of the corresponding DVN in terms of the Aggregated Flexibility of the participant DVNs through this equation:

$$Ratio_{DVN} = \frac{AmountFlexibility_{DVN} \cdot Reliability_{DVN}}{AF}$$

where:
$Ratio_{DVN}$: This ratio represents the ratio of the DVN's flexibility combined with its reliability in relation with the *AF*. Additionally this ratio reflects the ratio of the contribution of each DVN in case of a Demand Response signal.

This ratio expresses the percentage of the contribution of each DVN to the total Demand, while the Contribution of each DVN is expressed from the following equation:

$$Contribution_{DVN} = DemandAmount \cdot Ratio_{DVN}$$

where:

$DemandAmount$: Expresses the total demanded amount of Power Deviation.
$Contribution_{DVN}$: Expresses the demanded amount of Power deviation of a corresponding DVN

In case that the final extracted contribution of the respective DVN exceeds its total flexibility capacity, a mechanism is activated to distribute this excess of the corresponding asset to the remainder assets through the max min fairness algorithm.

$$ExcessDemand_{DVN} = Contribution_{DVN} - AmountFlexibility_{DVN}$$

$$CR = CR + ExcessDemand_{DVN}$$

Where:

$ExcessDemand_{DVN}$: The excess of Power that is demanded to be delivered from a specific DVN
CR: Represents the excess of energy that should be delivered from the remainder DVNs in order to find all the resources that match the demand.

This process repeats recursively generating some possible scenarios:

- The rest of the assets can satisfy the excess demand.
- One of the assets cannot serve the excess demand and the inner excess is shared among the rest of the assets. This process runs recursively, until the moment that there are no remainder DVNs.
- The scenario that demands cannot be satisfied from the participant DVNs and aggregator imports external DVNs.

5.2 Optimal Participation in Energy Markets

The DELTA aggregator is responsible to handle assets such as renewable energy sources, battery storages, and flexible consumption in order to make bidding decisions in markets with uncertain prices. Naturally, there is a high degree of uncertainty involved concerning the electricity generation of renewables, the flexibility of consumption and most importantly the prices in the electricity markets. Thus, we propose a multi-stage stochastic optimization approach for optimal participation in day ahead, intra-day and imbalance markets as described in chapter 2. In its first formulation, the proposed algorithm results to be a convex optimization scheme.

Stochastic optimization implies that uncertainty is dealt with by employing different scenarios with a corresponding probability of occurrence, so as to maximize the aggregator's profit over a whole set of possible scenarios. In the following subsections we further deploy the details of the proposed algorithm, together with the necessary constraints that describe the set of feasible solutions.

5.2.1 Two-stage stochastic optimization

Given the nature of the participation in the electricity markets problem, a two-stage (multi-stage in practice) stochastic optimization approach is proposed. A stochastic optimization approach implies that the uncertainty related to RES generation, flexibility and electricity prices is considered by employing scenarios which represent several realizations of the related random variables. A two-stage stochastic approach considers two kinds of variables: first stage and second stage variables. First stage variables are associated with decisions to be made before random variables take values and second stage variables are associated with decisions to be made depending on the first stage decisions and realizations of random variables. Thus, the two-stage stochastic optimization aims to find an optimal solution which includes decisions made in consecutive sessions in different markets and conditions.

A generic form of the objective function for a two-stage stochastic optimization problem is:

$$minimize f(X) + E[g(X, Y, \xi)],$$

where X is the set of first stage variables which are not scenario dependent variables. On the other hand, Y is the set of second stage variables, which are scenario dependent variables. Lastly, ξ is the set of random variables and the operator E computes the expected value of function g. The problem of optimal participation in the electricity markets can be formulated as a two-stage stochastic optimization problem by considering as first stage variables the decision variables that are to be made in a current market session and as second stage variables the decision variables that are to be made in future sessions.

5.2.2 Objective Function

For the purpose of problem formulation we form the list of related variables, which are presented in the following table.

Table 4. List of symbols and variables

Sets and subindex		***Decision variables***	
S	set of scenarios	P_t^{res}	RES power actually used (MW)
s	subindex of scenarios, $s=1,...,N_s$	E_t^{ess}	Energy stored (MWh)
t	subindex of time slot, $t=1,...,T$	$P_t^{ess,in}$	Power charging the storage (MW)
Parameters		$P_t^{ess,out}$	Power delivered by the storage (MW)
T	number of periods, let it be 24	$P_{s,t}$	Power to/from the aggregator (MW)
N_s	number of scenarios	P_t^{dam}	Power committed in DAM (MW)
ρ_s	probability of scenario s	P_t^{idm}	Power committed in IDM (MW)
P^{ess}	maximum power to/from battery	Δ_t^{im}	Deviation in IM (MW)
Random variables			
$\beta_t^{dam}, \beta_{s,t}^{dam}$	energy price in DAM (€/MWh)		
$\beta_t^{idm}, \beta_{s,t}^{idm}$	energy price in DAM (€/MWh)		
$\lambda_t^{im}, \lambda_{s,t}^{im}$	energy price in IM (€/MWh)		
$P_{s,t}^{res}$	RES power available (MW)		

The maximization of the aggregator's profit, as if a single IDM session is considered, can be expressed as

$$maximize \sum_{t=1}^{T} \left(\beta_t^{dam} \cdot P_t^{dam} + \beta_t^{idm} \cdot P_t^{idm} + \lambda_t^{im} \cdot \Delta_t^{im} \right)$$

The first two terms refer to the net income for participating in DAM and IDM, whereas the third term accounts for the implications of deviation with respect to the commitments acquired in DAM and IDM.

These deviations are handled by buying/selling energy in the imbalance market (IM). The proposed optimization scheme can be also expressed as a minimization problem as follows:

$$minimize\ -\sum_{t=1}^{T}\left(\beta_t^{dam}\cdot P_t^{dam}+\beta_t^{idm}\cdot P_t^{idm}+\lambda_t^{im}\cdot\Delta_t^{im}\right)$$

As it was mentioned previously, energy price for all markets, RES generation and flexibility of consumption are random variables. In order to deal with the uncertainty that is implied by those variables, we need to define a set of scenarios with an associated probability of occurrence, which is denoted as ρ_s. Finally, the optimization scheme, considering stochastic programming can be expressed as

$$minimize\ -\sum_{s=1}^{N_s}\rho_s\cdot\left(\sum_{t=1}^{T}\left(\beta_{s,t}^{dam}\cdot P_t^{dam}+\beta_{s,t}^{idm}\cdot P_t^{idm}+\lambda_{s,t}^{im}\cdot\Delta_t^{im}\right)\right)$$

It should be noted that the multiplicity of stages of the stochastic programming optimization is not apparent, because it is not actually considered in the expressed scheme above.

5.2.3 Modeling flexibility

As it was thoroughly described in chapter 3, there are multiple control policies that are applicable on demand response strategies. In the context of the DELTA project, there are basically two policies that are under consideration, which facilitate DR strategies for different types of DR requests and types of final users. First, a form of direct control policy and second a form of incentive-based indirect control policy are considered. Both approaches are described in the following subsections.

5.2.3.1 Direct control policy

In direct control policy considered in the context of the DELTA project, several assumptions are made in order to facilitate the formulation of the problem. Specifically, in this approach the flexible approach is assumed to be a portion of the total consumption that is always available to be reduced on demand of the aggregator. In other words, in this case the aggregator is able to request load dispatch for a specific amount of power from a DVN. The actual amount of flexible consumption from a certain DVN can additionally be characterized by a degree of reliability, depending on the historical ability of that particular DVN to deliver an amount of flexible power. The behaviour of this kind of flexibility is better illustrated in the following figure, where different degrees of reliable flexibility are represented by different colours in the bottom of the plot.

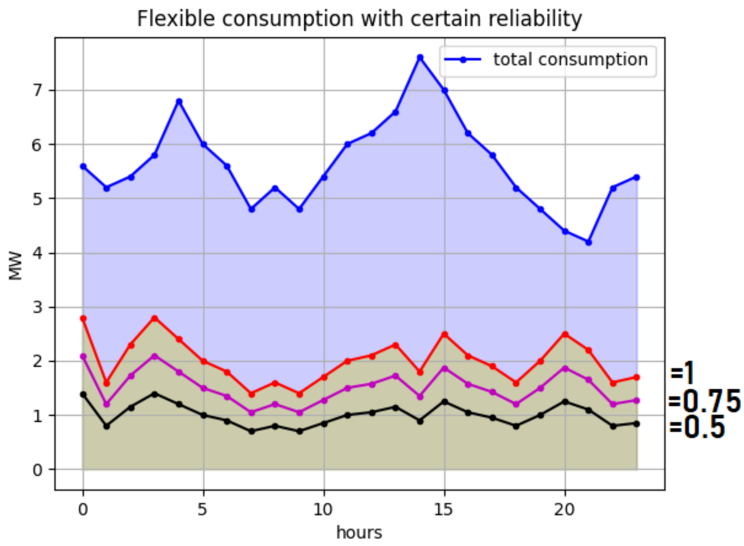


Figure 13. Example of flexible consumption with corresponding reliability

It should be noted that for practical reasons, trading flexible consumption is profitable only in periods where selling price of electricity in real-time market is high enough, for example higher than the retail market price.

Although modelling flexibility in the way that it was described in this subsection is convenient and easy to handle, it is nevertheless not very practical. Specifically, in the context of the DELTA project, information about which part of the consumption is flexible is not available. Instead, flexibility is an abstract measure of how much the final prosumers deviate from their usual consumption habits. Additionally, direct load dispatches are not always feasible in the DELTA approach. To conclude, a direct control policy is useful, yet it is not sufficient to model all internal demand response tasks performed into the DELTA Virtual Power Plant infrastructure.

5.2.3.2 Indirect control policy using Reinforcement Learning

To overcome the practical obstacles of direct control policy for demand-side management described in the previous paragraph, one should examine the alternative of an indirect control policy. In this case, the aggregator issues financial incentives to the final users using a signal, via the DVNs, in order to alternate their consumption behaviour. The reaction of the final users to the incentives given is a stochastic phenomenon and thus the aggregator is not able to develop a straight forward optimal strategy. To deal with this problem Reinforcement Learning has been proposed in the literature as a technique that can be used to train the aggregator to follow an optimal incentive policy that would maximize its profits [25]. In the context of the DELTA project, Reinforcement Learning, as in the following figure, has been proposed and is being examined so as to deal with optimal demand-side management. In the following paragraphs, the main components of the Reinforcement Learning approach considered in the DELTA project are deployed.

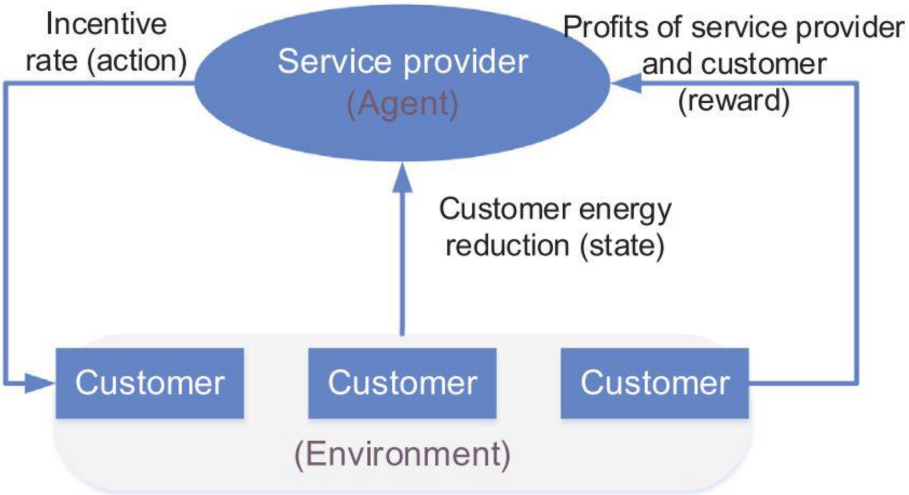


Figure 14. Reinforcement Learning scheme

Aggregator-agent model: In the DELTA architecture, the aggregator is interconnected with the DVN layer and thus direct communication with the final users as in [25] is not feasible. As a result, its uncertain environment is a number of DVNs which play the role of aggregated customers. Naturally, the objective is to find the optimal incentive $\lambda_{n,t}$ for $n = 1. \dots, N$, where N denotes the total number of DVNs and $t = 1, \cdots, T$, in order to maximize its profit. Assuming that the aggregator is planning to trade the reduced power to the real-time wholesale market, that is the imbalance market, its objective is to maximize its revenue in that particular market while minimizing the incentive payments to the customers-DVNs. This can be expressed as:

$$maximize \sum_{n=1}^{N} \sum_{t=1}^{T} (\lambda_t^{im} \cdot \Delta E_{n,t} - \lambda_{n,t} \cdot \Delta E_{n,t})$$

$$\lambda_{min} \leq \lambda_{n,t} \leq \lambda_{max}$$

where λ_t^{im} denotes the price from the imbalance market at hour t, $\Delta E_{n,t}$ and $\lambda_{n,t}$ are the demand reduction offered by, and incentive rate paid to the n^{th} DVN at hour t. In the inequality, λ_{min} and λ_{max} are the lower and upper bounds of incentive rate $\lambda_{n,t}$, which in practice can be decided by a contract between the aggregator and the final users, so as to protect both sides' profit.

Customers' model - DVNs' model: When informed of the incentive rate by the aggregator via the corresponding DVN, each final customer tries to maximize its incentive incomes by decreasing its energy consumption. However, reducing consumption can cause discomfort for the customer, which is most commonly modeled as a dissatisfaction cost. Thus, the goal of the customer is to maximize a mixture of incentive income minus a dissatisfaction cost as:

$$maximize \sum_{t=1}^{T} [\rho \cdot \lambda_{n,t} \cdot \Delta E_{n,m,t} - (1 - \rho) \cdot \phi_{n,m,t}(\Delta E_{n,m,t})]$$

$$\Delta E_{n,m,t} = E_{n,m,t} \cdot \xi_t \cdot \frac{\lambda_{n,t} - \lambda_{min}}{\lambda_{min}}$$

$$\mathrm{K}_{min} \leq \Delta E_{n,m,t} \leq \mathrm{K}_{max}$$

The first term in the maximization expression denotes the incentive income of the m^{th} customer under the n^{th} DVN at hour t by providing demand reduction $\Delta E_{n,m,t}$. The second term represents the dissatisfaction cost of that particular customer for the corresponding demand reduction. $\rho \in [0,1]$ is a weighting factor indicating the relative importance between the customer incentive income and discomfort. In the second equation, the variable $\Delta E_{n,m,t}$ denotes the demand reduction amount of the m^{th} customer under the n^{th} DVN at hour t, in which $E_{n,m,t}$ denotes indicates the energy demand of the same customer at that time slot. The variable ξ_t is the elasticity coefficient at hour t that denotes the ratio of energy demand change to incentive ratio variation. K_{min} and K_{max} are the lower and upper bounds for demand reduction.

The dissatisfaction cost function $\phi_{n,m,t}(\Delta E_{n,m,t})$ represents the degree of discomfort that a customer may experience when decreasing its energy demand. It is defined to be convex and it is expressed as:

$$\phi_{n,m,t}(\Delta E_{n,m,t}) = \frac{\mu_{n,m}}{2} \cdot \left(\Delta E_{n,m,t}\right)^2 + \omega_{n,m} \cdot \Delta E_{n,m,t}$$

The parameters $\mu_{n,m} > 0$ and $\omega_{n,m} > 0$ are dependent on the customer. Specifically, $\omega_{n,m}$is an auxiliary coefficient of the dissatisfaction cost function and the larger it is the higher the associated discomfort is. The parameter $\mu_{n,m}$ reflects the attitude of a customer with respect to electricity demand reduction. A larger $\mu_{n,m}$ implies that the customer is tolerant to less demand reduction and less discomfort.

As it is noted previously, in the DELTA architecture the aggregator has no access in the individual information of each customer. Instead it has access to the aggregated values of a DVN. For instance, the aggregated demand reduction is expressed as:

$$\Delta E_{n,t} = \sum_{m=1}^{M_n} \Delta E_{n,m,t}$$

where M_n denotes the number of prosumers that are under the n^{th} DVN. The optimization scheme in [25] includes both the profit of the aggregator and the profit on DVN side can be expressed as

$$\max \sum_{n=1}^{N} \sum_{t=1}^{T} \left\{\lambda_t^{im} \cdot \Delta E_{n,t} - \lambda_{n,t} \cdot \Delta E_{n,t} + \rho \cdot \lambda_{n,t} \cdot \Delta E_{n,t} - (1-\rho) \cdot \phi_{\mathrm{n,t}}(\Delta E_{n,t})\right\}$$

In the context of the DELTA project the behaviour of the final users is not considered known. As a result, the discomfort cost is not available to be used in the optimization of the aggregator.

Reinforcement Learning: RL is a machine learning algorithm allowing an agent to automatically determine the ideal behaviour in a stochastic environment, so as to maximize the cumulative reward [13].

RL is most commonly considered in the context of Markov decision process framework [26], which exhibits the Markov property that the state transitions are dependent only on the current state and current action taken, independently of all prior environmental stages or agent actions [25]. In the case of the DELTA architecture, we consider as reward the profit and as state the wholesale electricity price, which do not depend on the historical data. The key components that need to be modeled are a discrete time slot $t \in T$, a state $s_{n,t} \in S(\lambda_t^{\mathrm{im}})$, an action $a_{n,t} \in A(\lambda_{n,t})$ and a reward $r(s_{n,t}, a_{n,t}) \in R(\lambda_t^{\mathrm{im}}, \lambda_{n,t})$.

For the learning procedure we employ the Q-learning algorithm, which is a model-free off policy algorithm in RL [27]. In Q-learning we seek to find an optimal policy v^*, which in our case will be a sequence of incentive rates for each DVN and for each time slot, depending on the current energy demand reduction. The basic principle of Q-learning is the assignment of a Q-value $Q(s_{n,t}, a_{n,t})$ to each state-action pair at time $t \in T$, and updating of this value at each iteration in a manner that optimizes the result. In practice, in each time slot the agent performs an action, and the Q-value of the corresponding cell is updated as follows:

$$Q(s_{n,t}, a_{n,t}) \leftarrow (1-\theta) \cdot Q(s_{n,t}, a_{n,t}) + \theta \cdot [r(s_{n,t}, a_{n,t}) + \gamma \cdot maxQ(s_{n,t+1}, a_{n,t+1})] \,,$$

where $\theta \in [0,1]$ is the learning rate representing the degree at which the new override the old Q-values. A value of 0 implies that the agent learns nothing, exploiting prior knowledge exclusively, whereas a value of 1 implies that prior knowledge is totally ignored and only current estimation is taken under consideration. Naturally, there is a trade-off between learning from new experience and exploiting existing knowledge which the designer should take into account by properly selecting a value of θ between 0 and 1. The term $r(s_{n,t}, a_{n,t})$ represents the expected reward from a particular action at a certain state, which in our case is the expected profit of the customers and the aggregator. The variable $\gamma \in [0,1]$ indicates the relative importance of future versus present rewards.

In the Q-learning procedure, the agent directly interacts with the dynamic environment by executing actions. Then, the agent obtains a reward and moves to a new state, depending on the current state and action selected. The learning procedure is a result of trial and error during such iterations. After a sufficient number of iterations, the agent has interacted with the uncertain environment enough so as to have obtained an optimal policy. In other words, the Q-value gradually converges to a maximum. Since the Q-value represents the maximum reward with action $a_{n,t}$ at state $s_{n,t}$, the optimal policy is $v^* = argmaxQ(s_{n,t}, a_{n,t})$, which determines the optimal incentive rates at each state for a particular DVN.

5.2.4 Constraints

The proposed optimization scheme is subject to a number of constraints that define the feasible set of solutions. The complete optimization scheme considering a direct control policy as described above, including the objective function and the constraints can be expressed as:

$$minimize \; -\sum_{s=1}^{N_s} \rho_s \cdot \left(\sum_{t=1}^{T} \left(\beta_{s,t}^{dam} \cdot P_t^{dam} + \beta_{s,t}^{idm} \cdot P_t^{idm} + \lambda_{s,t}^{im} \cdot \Delta_t^{im} - \lambda_t^{ret} \cdot P_t^{flex} \right) \right)$$

$$s.t \; P_t^{flex} \geq 0$$

$$P_t^{flex} \leq \bar{P}_t^{flex}$$

$$E_t^{ess} = E_0^{ess} + \sum_{t=1}^{T} \eta_{in} P_t^{ess,in} - \sum_{t=1}^{T} \eta_{out} P_t^{ess,out}, \forall t \in T$$

$$E_0^{ess} = E_T^{ess}$$

$$E_t^{ess} \leq \bar{E}^{ess}$$

$$SOC_t = E_t^{ess} / \bar{E}^{ess}$$

$$P_t^{ess,out} \leq \bar{P}_{ess}$$

$$P_t^{ess,in} \leq \bar{P}_{ess}$$

$$P_{s,t}^{res} \leq \bar{P}_{s,t}^{res}$$

$$P_{s,t} = P_{s,t}^{res} + P_t^{ess,out} - P_t^{ess,in} + P_t^{flex}$$

$$\Delta_t^{im} = P_{s,t} - P_t^{dam} - P_t^{idm}$$

$$P_t^{ess,out}, P_t^{ess,in}, E_t^{ess} \geq 0$$

$$E_t^{ess} \leq \bar{E}^{ess}$$

$$-\bar{P}_{ess} \leq P^{dam} \leq \bar{P}_{ess} + \bar{P}_t^{res}$$

$$\left|P_t^{idm}\right| \overline{\leq P}_{ess} + \bar{P}^{res}, \forall t \in T, \forall s \in S$$

It should be noted that P_t^{flex} denotes the amount of flexible power reduction exploited from flexible consumption and λ_t^{ret} is the price of electricity in the retail market. Additionally, $\bar{P}_t^{flex}$ denotes the actual flexible consumption that is available in each time slot of the day.

6. Case Studies Implementation and Simulated experiments

For sake of comprehension it is useful to present a number of different case studies in order to demonstrate the behaviour of the proposed algorithm. In the following subsections we employ hypothetical data with corresponding forecasts in order to illustrate how the bidding decisions are made by the algorithm and what are the main challenges that are met in the context of the DELTA project.

For the needs of the simulations performed in this chapter we employ the definition of flexibility as it was described in subsection 5.2.3.1. Additionally, the optimization scheme presented in subsection 5.2.4 is used.

Moreover, we employ a scenario to demonstrate the Q-learning procedure that was described in subsection 5.2.3.2. Specifically, we illustrate the behaviour of the learning agent when seeking to optimize its incentive-based DR policy.

6.1 Forecasts and Actual Data

For the purposes of the simulations we employ artificial data. First, hypothetical "actual" prices, RES generation available and flexibility of consumption are assumed. Next, the corresponding forecasts are created by adding uncertainty, in the form of random noise of variable amplitude, on the initial data. At last, "noise" data is fed to the algorithm which will be finally evaluated on the initial "actual" data.

The values of the parameters of the aggregator's assets are considered to be $\bar{P}_{ess} = 5MW$, $\bar{E}^{ess} = 5MWh$, $E_0^{ess} = 2{,}5MWh$, $\eta_{in} = 0.9$, $\eta_{out} = 0.9$ and $SOC_{min} = 0.05$.

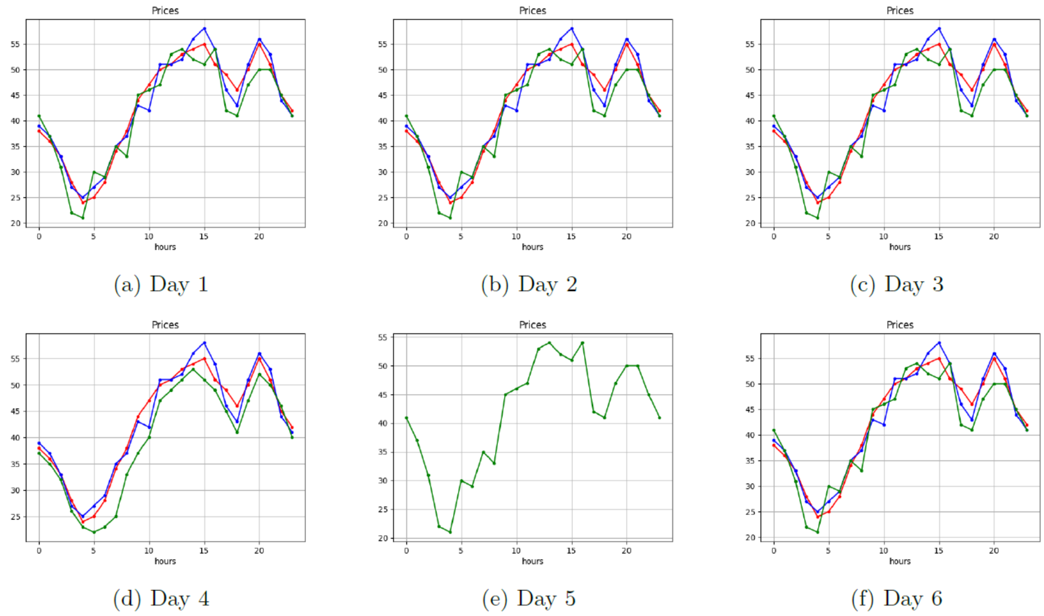
Since the optimization in each day is performed independently, it is useful to consider different scenarios for each day to better demonstrate the behaviour of the algorithm. In the following table we present the main characteristics of each day-scenario.

Table 5. Cases of the six days considered

Day	Basic scenarios for 6 days
1	No RES
2	No flexibility
3	No flexibility, No RES, Bad forecasts in IM.
4	No flexibility, No RES, Worse forecasts in IM, different prices
5	Participation only in IM, No RES
6	Fine forecasts, RES generation, flexibility

For the first day we consider a regular set of market prices together with some flexibility of consumption, yet we assume no RES generation. In the case of the second day, same set of prices is considered and no flexibility of consumption. These first two cases are useful in order to demonstrate how profits fluctuate and how the algorithm reacts when having different combination of assets available. In day 3 we consider the combination of the previous two cases, together with noisy forecasts with respect to the imbalance market. Moreover, in the fourth day we employ a slightly differentiated set of prices and assume even more increased uncertainty in the prices of the imbalance market. This case study is used to illustrate the risk of participating in the electricity markets with unreliable forecasts. In the case of the fifth day, we constrain participation in only IM to illustrate how the aggregator can profit when market arbitrage is not feasible. Finally, for the sixth day we select favourable conditions for the aggregator to demonstrate the profits when all assets are available and forecasts succeed to capture the behaviour of actual prices.

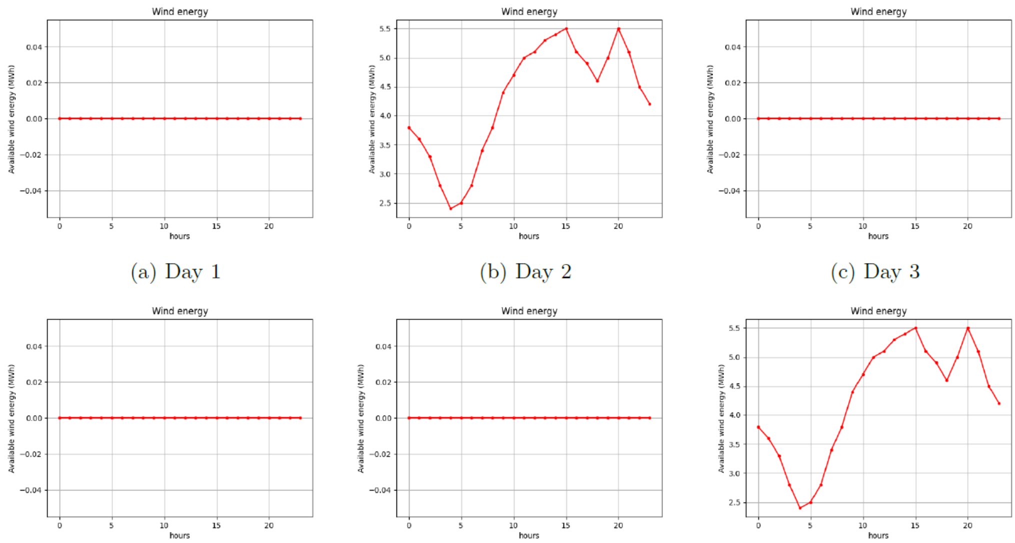
In the following figure one can see the "actual" values for the prices in the three markets for all six days of this simulation. It is noted that the red line represents the DAM prices, blue line represents the IDM prices and finally the green line describes the prices in IM.



(a) Day 1 (b) Day 2 (c) Day 3
(d) Day 4 (e) Day 5 (f) Day 6

Figure 15. Actual prices in six days (red: DAM, blue: IDM, green: IM)

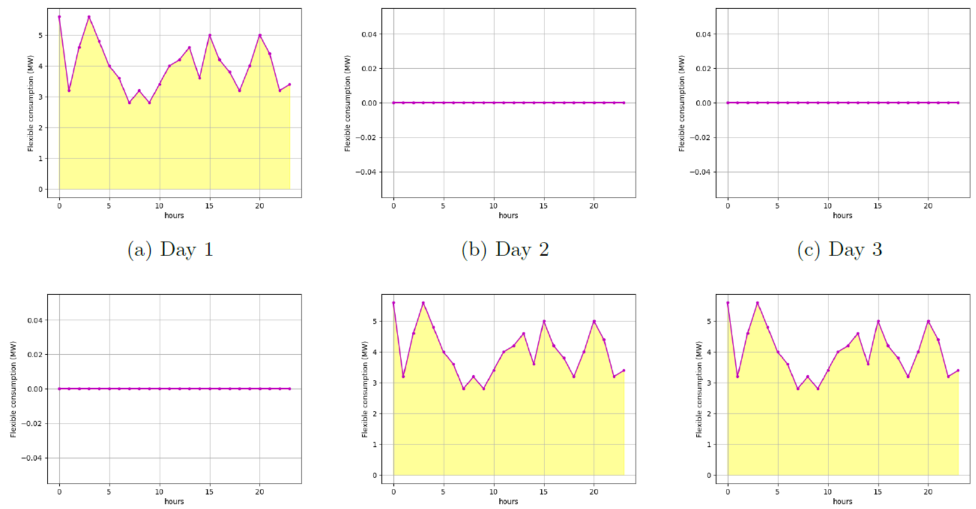
In the following figure one can see the "actual" values for the available RES generation and in the one after that, the corresponding values for available flexibility.



(a) Day 1 (b) Day 2 (c) Day 3

(d) Day 4 (e) Day 5 (f) Day 6

Figure 16. RES available in each of six days

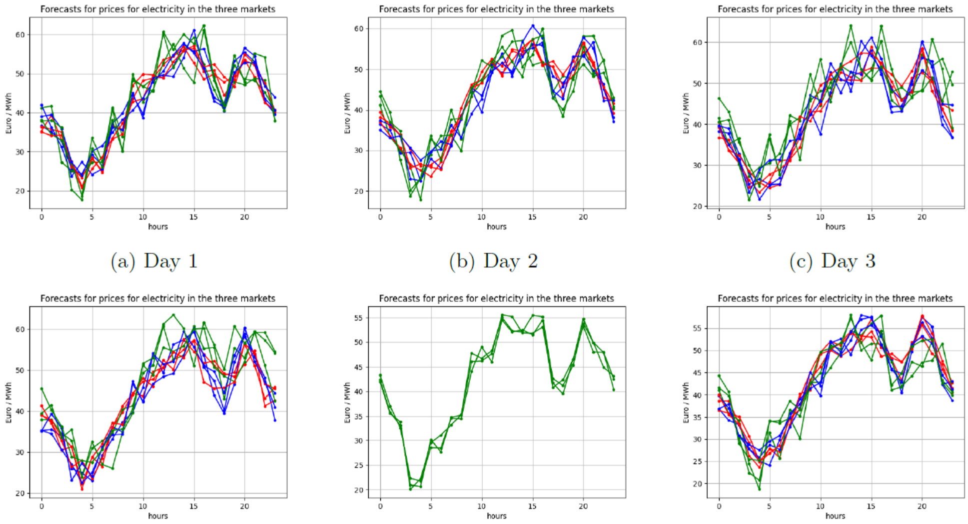


(a) Day 1 (b) Day 2 (c) Day 3

(d) Day 4 (e) Day 5 (f) Day 6

Figure 17. Flexible consumption available in each of six days

As noted previously, forecast and scenario generation is a crucial part of the stochastic optimization. For the purposes of the current simulations we use artificially created forecasts. In the following figure the scenarios generated for market prices of each day are represented.



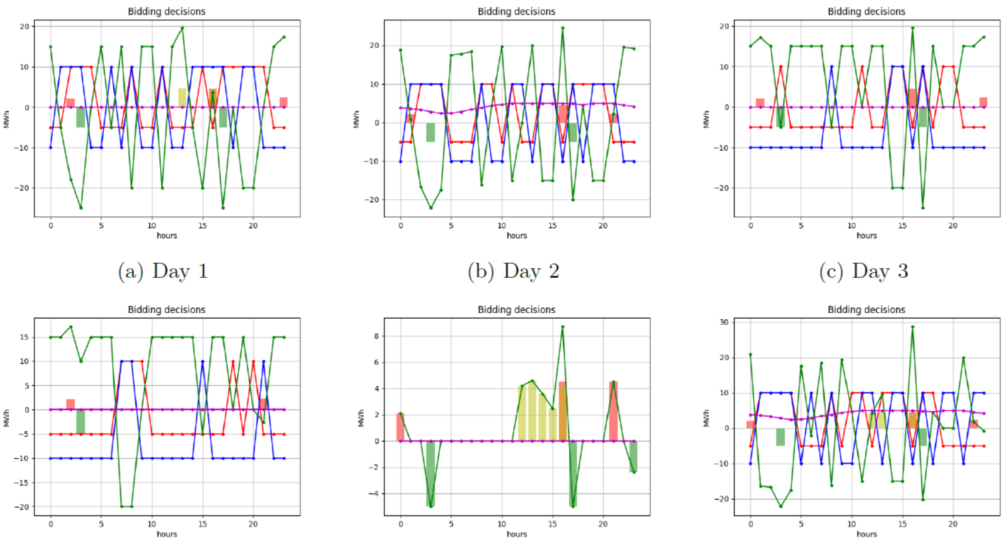
(a) Day 1 (b) Day 2 (c) Day 3

(d) Day 4 (e) Day 5 (f) Day 6

Figure 18. Forecasted prices in six days (red: DAM, blue: IDM, green: IM)

6.2 Results

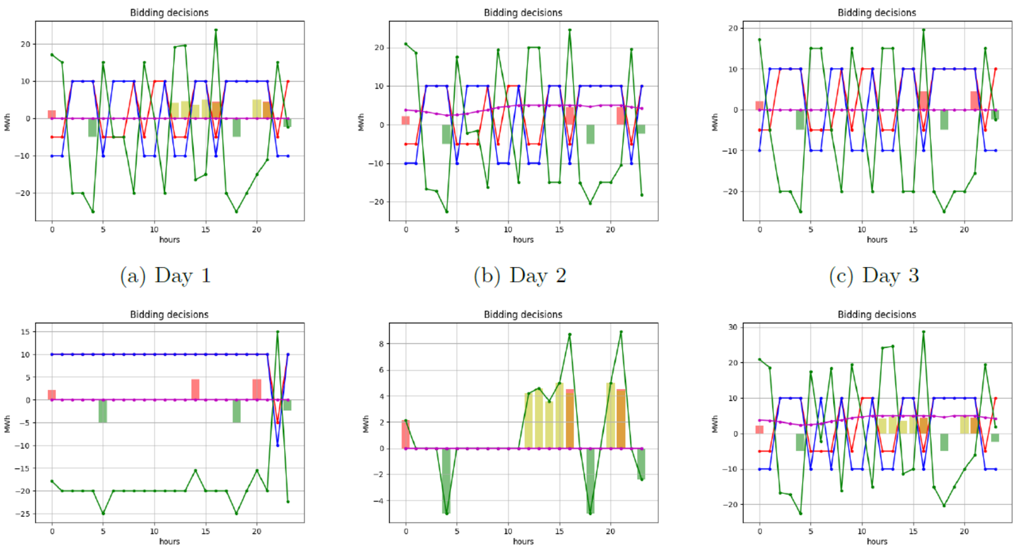
In this subsection the bidding decisions are presented for the cases described in the previous subsection. Specifically, in the following figure the decisions of the stochastic optimization are presented. In the figure after that, the corresponding bidding decisions considering a perfect information hypothesis are presented.



(a) Day 1 (b) Day 2 (c) Day 3

(d) Day 4 (e) Day 5 (f) Day 6

Figure 19. Bidding decisions considering uncertainty



(a) Day 1 (b) Day 2 (c) Day 3

(d) Day 4 (e) Day 5 (f) Day 6

Figure 20. Bidding decisions considering perfect information

In the following two tables we present the profits that result from the proposed optimization and given the data for the six days, as described previously. Specifically, we present the profit under uncertainty and the hypothetical profit in the case that perfect information is available. It is noted that the profit of the aggregator is calculated according to the following expression.

$$Profit = \sum_{t=1}^{T} \left(\beta_t^{dam} \cdot P_t^{dam} + \beta_t^{idm} \cdot P_t^{idm} + \lambda_t^{im} \cdot \Delta_t^{im} - \lambda_t^{ret} \cdot P_t^{flex} \right)$$

Table 6. Net income per market considering uncertainty

Net income (€) - considering Uncertainty				
Day	**DAM**	**IDM**	**IM**	**Total**
1	1213	905	-954	1164
2	2759	2813	-194	5578
3	-1954	-4700	7104	450
4	-2510	-7799	9980	-329
5	0	0	216	216
6	2140	120	3176	5436

Table 7.Net income per market considering prefect information

Net income (€) - considering Perfect Information				
Day	**DAM**	**IDM**	**IM**	**Total**
1	4195	1500	-4234	1460
2	4195	420	1132	5747
3	4869	420	-3920	1369
4	9594	9419	-17326	1688
5	0	0	240	240
6	4869	1159	-191	5838

As it can be seen from profits of day 4, forecasts' failure to capture the behaviour of prices in the electricity market could lead to negative profits. In other words, forecast failure might lead to losses. Additionally, from results of day 5 it can be noticed that exclusive participation in IM does not allow high profits. This is reasonable given that in this case market arbitrage strategies are not feasible. At last, profits being much higher on day 2 and day 6 is due to the fact that in those two days RES generation is considered and thus more energy is sold to the markets

6.3 Q-learning for indirect control policy

In this subsection, we present an example of the Q-learning algorithm for the purpose of optimal incentive-based DR strategy. The entire procedure seeks to find what is the optimal financial incentive given to a certain DVN based on the real-time wholesale electricity price, in order to maximize the net income of the aggregator. By contrast to the approach in [25] where the consumer behaviour is considered known, the DELTA aggregator assumes no knowledge about the DVNs' reaction to certain incentive. The learning procedure is better illustrated by the following algorithmic steps.

1. Initialize incentive rate bounds, demand reduction ranges, dissatisfaction cost parameters and weighting factor ρ of each customer within a DVN.
2. Initialize $t \leftarrow 0$ and the Q-value $Q^t(s_{n,t}, a_{n,t}) \leftarrow 0$
3. **Do:**
4. Observe the wholesale electricity price $s_{n,t} \in S(\lambda_t^{im})$
5. Select an incentive rate $a_{n,t} \in A(\lambda_{n,t})$ by ϵ-greedy policy
6. Observe the demand reduction reaction of each DVN, $\Delta E_{n,t}$
7. Compute the reward $r(s_{n,t}, a_{n,t}) \leftarrow (\lambda_t^{im} - \lambda_{n,t}) \cdot \Delta E_{n,t}$
8. Update the Q-value: $Q(s_{n,t}, a_{n,t}) \leftarrow (1-\theta) \cdot Q^t(s_{n,t}, a_{n,t}) + \theta \cdot r(s_{n,t}, a_{n,t})$
9. $t \leftarrow t + 1$
10. **While** $|Q^t - Q^{t-1}| \geq \epsilon$
11. Optimal policy is: $v^* = argmaxQ^t(s_{n,t}, a_{n,t})$

The demand reduction that is provided by each consumer as a reaction to a certain financial incentive is the result of each final user's consuming behaviour. Thus, $\Delta E_{n,t}$ is modelled as the result of the following optimization scheme for each consumer/DVN:

$$maximize \sum_{t=1}^{T} [\rho_n \cdot \lambda_{n,t} \cdot \Delta E_{n,t} - (1-\rho_n) \cdot \phi_{n,t}(\Delta E_{n,t})]$$

$$\mathrm{K}_{min} \leq \Delta E_{n,t} \leq \mathrm{K}_{max}$$

As described in subsection 5.2.3.2. It should be highlighted that the real-time wholesale electricity price λ_t^{im} for the current hour is not known in the general case. Instead, as described in the previous section the participation in the imbalance market is performed by considering a forecasting mechanism. Yet, for the purposes of this subsection we consider real-time price to be known so as to demonstrate the behaviour of the proposed Q-learning algorithm

6.3.1 Single customer DVN

In this subsection we employ a hypothetical DVN which consists of a single customer so as to demonstrate how its specific characteristics affect the resulting demand reduction and of course the resulting profit of the aggregator. For the purposes of this simulation consider a customer with the following characteristics

CU_1	$\mu_1 = 5$	$\omega_1 = 5$	$K_{min} = 0$	$K_{max} = 0.3 \cdot E_{n,t}$

For the learning procedure we need to specify what is the observation space and the action space. For this example we consider that $\lambda_t^{im} \in [15,50]$ and $\lambda_{n,t} \in [5,25]$, which denote currency per MWh. Additionally, for both state and action spaces we consider a quantization and a corresponding discretization in 10 values for each. As a result the Q-table will be a 10×10 table. Considering training over 500 iterations/episodes, $\epsilon = 0.9$ and some details about the Q-table can be illustrated in the following figure.

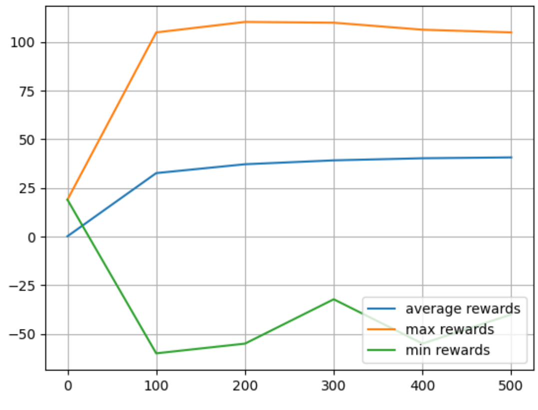


Figure 21 Learning curve for ρ_1=0.5

As it can be seen, 200 episodes are enough for the Q-values to reach a plateau. In order to demonstrate the behaviour of the trained agent we employ some hypothetical real-time price data and a corresponding hypothetical demand curve. The resulting behaviour is illustrated in the following figures for three different cases of ρ_1 value. The plots in blue describe the demand curve considering no financial incentives given to the customer and the plots in red describe the reduced consumption.

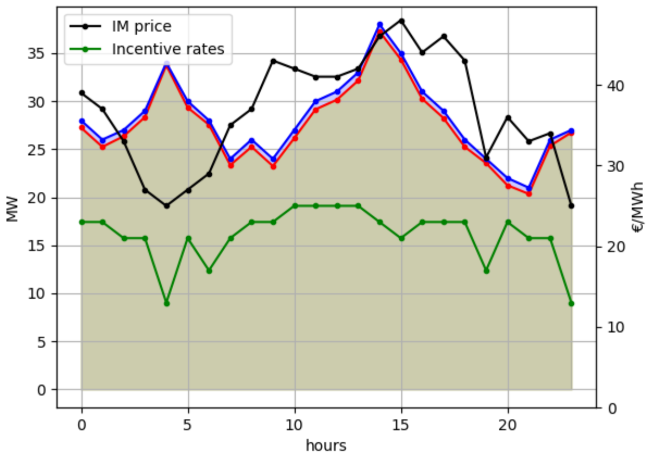


Figure 22 ρ_1=0.1

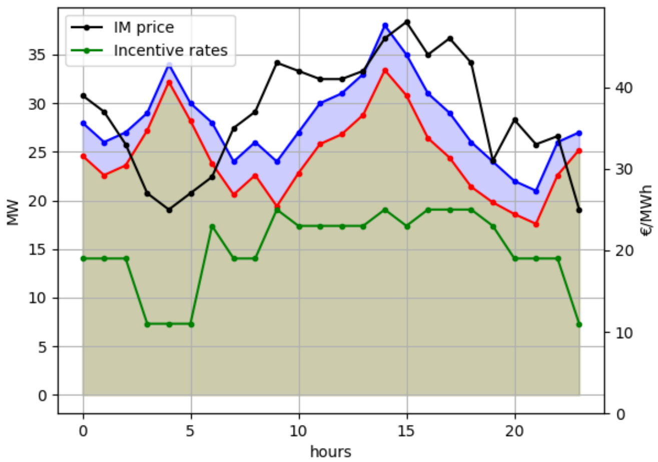


Figure 23. ρ_1=0.5

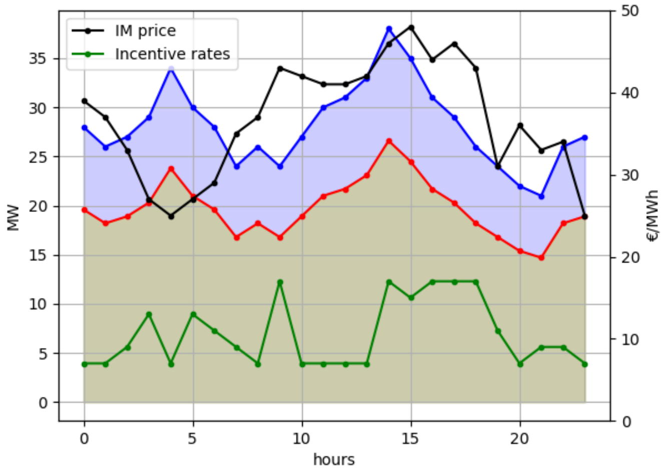


Figure 24 ρ_1=0.9

The resulting profits of the aggregator for each of the three customer cases are demonstrated in the following table.

	Aggregator's profit	Customer's profit
$\rho_1 = 0.1$	255 €	367 €
$\rho_1 = 0.5$	1444 €	1108 €
$\rho_1 = 0.9$	5351€	2492 €

As it can be seen, the value of the variable ρ represents the willingness of a certain customer to participate in the DR program. In other words, for a customer with a larger value of ρ it is of higher priority to maximize its profits from the DR program than to minimize its discomfort. Naturally, the same approach can be extended to a case of a DVN consisting of multiple customers.

7. Conclusions

This report presented in detail the architectural design and implementation of the Decision Support System of the DELTA Aggregator. Targeting a more dynamic behaviour that can adapt in various markets within the day while also servicing other existing contracts. To achieve this the DELTA Aggregator has been developed with two core engines, one for servicing any direct incoming DR request, for example from the DSO due to another existing contract, and the second to identify and optimize the Aggregator's participation to the day-ahead, intra-day and imbalance markets. For these two core components, a different implementation flow has been adopted, following cutting edge technologies.

Towards that direction, and focusing on leveraging on multiple dynamic markets, further effort was denoted towards addressing uncertainty, which is inherent in the problem of optimal participation in the electricity markets, by expanding the forecasting market price engine developed through T4.3 to cover also the day-ahead and intra-day markets. Additional progress on the Net Imbalance Volume has been introduced as well.

Building upon and enhancing other components, the DELTA DSS covers mechanisms for both instant DR and optimal participation to the DR markets. Investing on a two-stage stochastic optimisation approach which is enriched by a reinforcement learning technique, the DELTA Aggregator can "navigate" successfully through the different markets and delivers new revenue streams for both the Aggregator and the Customers.

As the integration is finalised and the pilot deployment commences, it is expected to have further development of the Decision Support System of the DELTA Aggregator. For instance, as discussed in chapter 3, introducing indirect control policies for implicit DRs requires employing novel techniques such as reinforcement learning. Such improvements of the capabilities of the DELTA aggregator are part of the refinement that is necessary for the testing and final integration of the project. As such, the aforementioned updates will be further described in future reports.

References

[1]. Conejo, Antonio J., Miguel Carrión, and Juan M. Morales. *Decision making under uncertainty in electricity markets*. Vol. 1. New York: Springer, 2010.

[2]. Crespo-Vazquez, Jose L., et al. "A machine learning based stochastic optimization framework for a wind and storage power plant participating in energy pool market." *Applied Energy* 232 (2018): 341-357.

[3]. Cramton, Peter, Axel Ockenfels, and Steven Stoft. "Capacity market fundamentals." *Economics of Energy & Environmental Policy* 2.2 (2013): 27-46.

[4]. Greenwood, D. M., et al. "Frequency response services designed for energy storage." *Applied Energy* 203 (2017): 115-127.

[5]. Vardakas, John S., Nizar Zorba, and Christos V. Verikoukis. "A survey on demand response programs in smart grids: Pricing methods and optimization algorithms." IEEE Communications Surveys & Tutorials 17.1 (2014): 152-178.

[6]. Palensky, Peter, and Dietmar Dietrich. "Demand side management: Demand response, intelligent energy systems, and smart loads." IEEE transactions on industrial informatics 7.3 (2011): 381-388.

[7]. Kosek, Anna Magdalena, et al. "An overview of demand side management control schemes for buildings in smart grids." 2013 IEEE international conference on smart energy grid engineering (SEGE). IEEE, 2013.

[8]. Strbac, Goran. "Demand side management: Benefits and challenges." Energy policy 36.12 (2008): 4419-4426.

[9]. Alagoz, B. B., A. S. İ. M. Kaygusuz, and A. Karabiber. "A user-mode distributed energy management architecture for smart grid applications." Energy 44.1 (2012): 167-177.

[10]. Tanenbaum, Andrew S., and Maarten Van Steen. Distributed systems: principles and paradigms. Prentice-Hall, 2007.

[11]. Heussen, Kai, et al. "Indirect control for demand side management-A conceptual introduction." 2012 3rd IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe). IEEE, 2012.

[12]. Gehrke, Oliver, and Fridrik Isleifsson. "An aggregation friendly information model for demand side resources." IEEE Local Computer Network Conference. IEEE, 2010.

[13]. Panapakidis, Ioannis P., and Athanasios S. Dagoumas. "Day-ahead electricity price forecasting via the application of artificial neural network based models." Applied Energy 172 (2016): 132-151.

[14]. Lago, Jesus, et al. "Forecasting day-ahead electricity prices in Europe: the importance of considering market integration." Applied energy 211 (2018): 890-903.

[15]. de Menezes, Lilian M., and Melanie A. Houllier. "Reassessing the integration of European electricity markets: A fractional cointegration analysis." Energy Economics 53 (2016): 132-150.

[16]. Conejo, Antonio J., et al. "Day-ahead electricity price forecasting using the wavelet transform and ARIMA models." IEEE transactions on power systems 20.2 (2005): 1035-1042.

[17]. Tan, Zhongfu, et al. "Day-ahead electricity price forecasting using wavelet transform combined with ARIMA and GARCH models." Applied energy 87.11 (2010): 3606-3610.

[18]. Girish, Godekere P. "Spot electricity price forecasting in Indian electricity market using autoregressive-GARCH models." Energy Strategy Reviews 11 (2016): 52-57.

[19]. Khandelwal, Ina, Ratnadip Adhikari, and Ghanshyam Verma. "Time series forecasting using hybrid ARIMA and ANN models based on DWT decomposition." Procedia Computer Science 48.1 (2015): 173-179.

[20]. Yang, Zhang, Li Ce, and Li Lian. "Electricity price forecasting by a hybrid model, combining wavelet transform, ARMA and kernel-based extreme learning machine methods." Applied Energy 190 (2017): 291-305.

[21]. Mirakyan, Atom, Martin Meyer-Renschhausen, and Andreas Koch. "Composite forecasting approach, application for next-day electricity price forecasting." Energy Economics 66 (2017): 228-237.

[22]. Chitsaz, Hamed, et al. "Electricity price forecasting for operational scheduling of behind-the-meter storage systems." IEEE Transactions on Smart Grid 9.6 (2017): 6612-6622.

[23]. Kath, Christopher, and Florian Ziel. "The value of forecasts: Quantifying the economic gains of accurate quarter-hourly electricity price forecasts." Energy Economics 76 (2018): 411-423.

[24]. Ke, Guolin, et al. "Lightgbm: A highly efficient gradient boosting decision tree." Advances in neural information processing systems. 2017.

[25]. Lu, Renzhi, and Seung Ho Hong. "Incentive-based demand response for smart grid with reinforcement learning and deep neural network." Applied energy 236 (2019): 937-949.

[26]. Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

[27]. Watkins, Christopher JCH, and Peter Dayan. "Q-learning." Machine learning 8.3-4 (1992): 279-292.