

# Feature-driven trading of wind power and hydrogen



**Emil Helgren**  
**DTU Wind-M-0605**  
February 2023

**Author:** Emil Helgren

**DTU Wind-M-0605**  
**February 2023**

**Title:**  
Feature-driven trading of wind power and hydrogen

**Project period:**  
September 2022 - February 2023

**ECTS: 30**

**Education: Master of Science**

**Supervisor(s):**

Jalal Kazempour  
Leisa Mitridati  
**DTU Wind and Energy Systems**

**Remarks:**

This report is submitted as partial fulfillment of the requirements for graduation in the above education at the Technical University of Denmark.

DTU Wind and Energy Systems is a department of the Technical University of Denmark with a unique integration of research, education, innovation and public/private sector consulting in the field of wind and energy. Our activities develop new opportunities and technology for the global and Danish exploitation of wind and energy. Research focuses on key technical-scientific fields, which are central for the development, innovation and use of wind energy and provides the basis for advanced education.

**Technical University of Denmark**  
Department of Wind and Energy  
Systems Frederiksborgvej 399  
DK-4000 Roskilde  
[www.wind.dtu.dk](http://www.wind.dtu.dk)

## **Abstract**

This thesis investigates how data-driven models can be used to operate an electrolyzer unit in combination with a wind farm, from the perspective of a wind and hydrogen producer acting in the Danish electricity markets and a bilateral hydrogen market. Trading wind energy in a forward market involves decision-making under uncertainty, where deterministic or probabilistic forecasts are the usual methods used to produce the market bids. Data-driven models have shown to be a relevant alternative to the established methods, and are of even greater interest as the availability of data in the power sector increases. By introducing hydrogen production as a possibility, the operation of the electrolyzer and the forward market bid becomes competing sources of revenue based on the same produced power, meaning the problem becomes a task of choosing the allocation between the two sources resulting in the highest revenue. As the electrolyzer unit allows for real-time adjustment of the hydrogen production, a possibility for recourse decisions is introduced, and the problem of adjusting the electrolyzer according to the most recent available data becomes a new potential for improvement. The thesis demonstrates that data-driven models are capable of out-performing a deterministic approach in the task of maximizing the revenue gain from combined wind trading and hydrogen production. Furthermore, it is demonstrated that increased revenue can be gained by implementing deterministic adjustment algorithms, that adjust the hydrogen production in real-time based on the realized production.

## Acknowledgements

I would like to express my utmost gratitude to my supervisors Jalal Kazempour and Lesia Mitridati who guided me through the project, and have been an indispensable part of the entire process from the initial idea to the finished product. I would also like to thank Siemens Gamesa for allowing me to use their forecast data and thereby making the results more representative of a real-world application of the developed methods, and to Alice Patig & Bruno Laurini for cleaning and preparing the data for use. I would also like to extend thanks to Enrica Raheli for essential guidance regarding the electrolyzer modelling.

## Reader's Guide

The content of the report is given in 10 chapters, followed by an Appendix containing relevant but non-essential content related to the report. A list of tables and figures as well as an overview of the general nomenclature is found immediately after the table of contents. All citations and references to figures, equations, sections and tables are given as clickable hyperlinks.

The complete code base for the project can be found on GitHub:

<https://github.com/emilhelgren/feature-driven-trading-of-wind-power-and-hydrogen>.

All public data used in the project can be exported from [2].

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Litterature Review</b>	<b>4</b>
<b>3</b>	<b>Data</b>	<b>6</b>
3.1	Price Data . . . . .	6
3.2	Production Data . . . . .	11
3.3	Summary of Data Chapter . . . . .	14
<b>4</b>	<b>Electrolyzer Modelling</b>	<b>15</b>
4.1	Electrolyzer technology . . . . .	15
4.2	Conversion and Efficiencies . . . . .	16
4.3	Pricing and Production Quotas . . . . .	16
4.4	Further Modelling Simplifications . . . . .	17
<b>5</b>	<b>Model Development</b>	<b>18</b>
5.1	Mathematical Formulation . . . . .	18
5.2	The Deterministic Approach . . . . .	26
5.3	Feature-Driven Model . . . . .	28
5.4	The Feature Vector . . . . .	40
5.5	Final Models . . . . .	43
<b>6</b>	<b>Real Time Adjustment</b>	<b>49</b>
6.1	Adjustment Decision Rule . . . . .	49
6.2	Adjustment Policies . . . . .	52
<b>7</b>	<b>Benchmarks for Evaluation</b>	<b>58</b>
7.1	Hindsight . . . . .	58
7.2	Optimal Adjustment . . . . .	58
7.3	Static Oracle . . . . .	58
<b>8</b>	<b>Results</b>	<b>60</b>
8.1	Feature-driven Models . . . . .	60
8.2	Adjustment . . . . .	63
8.3	Deterministic Model . . . . .	65
8.4	Hydrogen Penalty . . . . .	66
8.5	Distribution of Revenue . . . . .	67
8.6	Testing in 2022 . . . . .	70
8.7	Balancing Settlements . . . . .	72
<b>9</b>	<b>Discussion</b>	<b>75</b>
<b>10</b>	<b>Conclusion</b>	<b>78</b>
<b>A</b>	<b>Appendix</b>	<b>i</b>

## List of Tables

1	Overview of constants . . . . .	18
2	Wasserstein distances between medoids . . . . .	i
3	Wasserstein distances between scaled medoids . . . . .	i

## List of Figures

1	Map of current electrolyzer projects in Denmark . . . . .	2
2	Summary of price forecast error distribution fittings . . . . .	7
3	Price forecast error distribution fitting . . . . .	8
4	Synthetic price forecast errors . . . . .	8
5	Mean price days 2019-2021 . . . . .	9
6	K-medoid clustering of price days - normalized to 2019 . . . . .	10
7	Schematic of price day distributions . . . . .	10
8	Wind power curve . . . . .	12
9	K-medoid clustering of production days . . . . .	12
10	Summary of production forecast error distribution fittings . . . . .	13
11	Production forecast error distribution fitting . . . . .	13
12	Synthetic production forecast errors . . . . .	13
13	Schematic of hydrogen production . . . . .	15
14	Timeline of electricity markets . . . . .	18
15	Different approaches to solving the base model . . . . .	25
16	Production bid in forward market . . . . .	28
17	Consumption bid in forward market . . . . .	28
18	Schematic of the feature-driven model . . . . .	29
19	Schematic of sliding window approach . . . . .	31
20	Example of a bidding curve in the forward market . . . . .	32
21	Linear function as model output . . . . .	33
22	Low resolution bidding curve . . . . .	33
23	High resolution bidding curve . . . . .	33
24	Price curve resulting from three price domains . . . . .	38
25	Schematic overview of <i>General</i> architecture . . . . .	39
26	Schematic overview of <i>Hourly</i> architecture . . . . .	39
27	Schematic of all features being exposed to a model . . . . .	40
28	Schematic of a forecast model combining the different features . . . . .	41
29	Overview of feature vectors . . . . .	42
30	Schematic overview of <i>General</i> architecture . . . . .	44
31	Schematic overview of <i>Hourly</i> architecture . . . . .	45
32	Schematic overview of <i>General</i> architecture with price domains . . . . .	46
33	Schematic overview of <i>Hourly</i> architecture with price domains . . . . .	47
34	Forward price branching . . . . .	50
35	High forward price scenario . . . . .	50
36	Low forward price scenario . . . . .	51
37	Complete decision rule . . . . .	51
38	Upwards Adjustment . . . . .	53
39	Produced and planned hydrogen production . . . . .	54
40	Upwards and Downwards Adjustment . . . . .	55
41	Results for models <b>GA</b> and <b>HA</b> . . . . .	60

42	Results for models <b>GAPD</b> , <b>HAPD</b> , <b>GAPDR</b> and <b>HAPDR</b> . . . . .	61
43	<b>HAPD-AF-12</b> with benchmarks . . . . .	63
44	Results of upwards adjustment algorithm . . . . .	64
45	Results of upwards and downwards adjustment algorithm . . . . .	65
46	Deterministic model with all bids accepted . . . . .	66
47	Impact of missing production penalty . . . . .	67
48	Distribution of revenue between hydrogen and forward market . . . . .	68
49	Total revenue with reduced electrolyzer size - <b>HAPD</b> . . . . .	69
50	Distribution of revenue with reduced electrolyzer size . . . . .	70
51	Results for model <b>HAPD</b> on 2022 data . . . . .	71
52	Distribution of revenue for 2022 data . . . . .	72
53	Balancing settlements overview . . . . .	73
54	Schematic of domain specific parameters multiplying . . . . .	76
55	Results of postponing adjustment algorithm . . . . .	ix
56	Enlarged overview of results . . . . .	x

---

## Nomenclature

### Operators

$(\cdot)^+$	The positive part of the argument, or equivalently: $\max(\cdot, 0)$
$(\cdot, \cdot)$	Open interval of real numbers
$[\dots]$	Closed interval of integers
$[\cdot, \cdot]$	Closed interval of real numbers
$\text{dom}(f)$	Domain of the function $f$
$\text{mod}(\cdot)$	Modulo operator
$\mathbf{a}^\top$	The transpose of vector $\mathbf{a}$

### Constants and variables

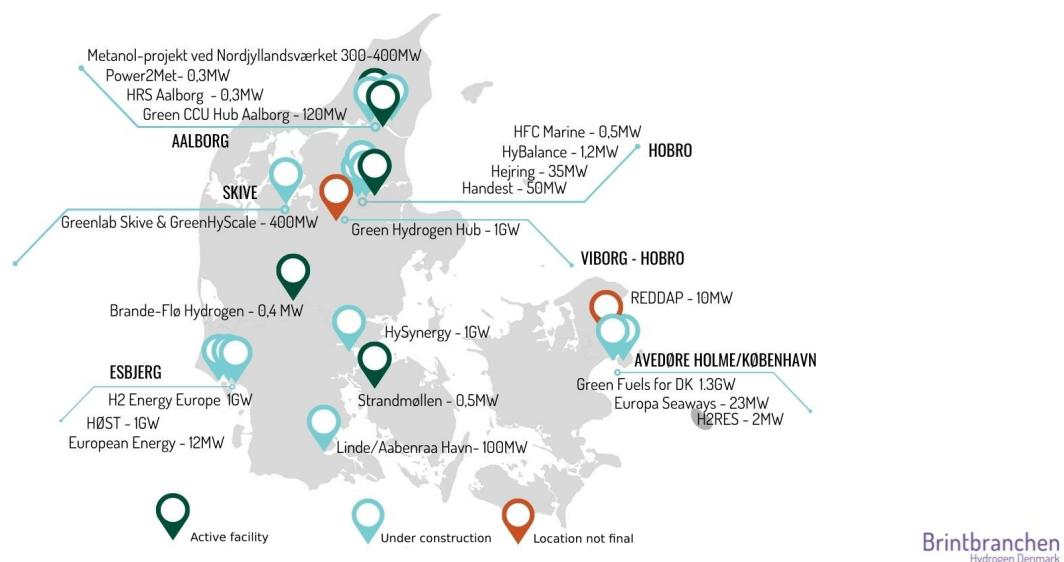
$\hat{\lambda}^F$	Forward price point forecast
$\hat{p}^{FC}$	Production point forecast
$\lambda^F$	Forward price
$\lambda^H$	Hydrogen price
$\lambda^{DW}$	Downwards balancing price
$\lambda^{UP}$	Upwards balancing price
$\overline{P}^F$	Rated wind capacity
$\overline{P}^H$	Rated electrolyzer capacity
$\psi$	Missing production penalty
$E^{real}$	Realized production
$P^{DMin}$	Daily hydrogen production requirement
$p^F$	Forward market bid
$p^H$	Hydrogen production allocation

## 1 Introduction

The transition to renewable energy sources has become a key focus for many governments and organizations around the world in an attempt to reduce the dependence on fossil fuels and combat climate change. One of the most important aspects of this transition is the electrification of residential appliances as well as various sectors of industry, so that the energy can be sourced from renewable production units instead of direct fossil fuel consumption.

Renewable energy sources like wind and solar are characterized by the fact that their production is uncertain and non-dispatchable, meaning that a high penetration of these units in the energy mix will result in periods of excess energy in the grid occurring more often. To avoid the loss of this energy due to curtailment of the production units, the excess production should either be stored for use at other times of insufficient supply, or used directly for another valuable purpose. Power-to-X (ptX) is a possible means to both of these ends. The concept of ptX is to direct surplus energy into an electrolyzer to produce hydrogen, which can then either be turned back into electricity by use of a fuel cell, or further distributed to the industry for either direct usage or as an intermediate step in the production of green fuels.

The Danish government have declared ambitious goals to increase hydrogen production capabilities, both for the purpose of utilizing surplus electricity production in the described manner, as well as entering the sector of green fuel production<sup>1</sup>. Figure 1 shows an overview of current electrolyzer projects in Denmark, where it is seen that more than 5 GW of electrolyzer capacity is either active or under construction. Also the European commission have stated an intention to build 17.5 GW of electrolyzer capacity by 2025 across the EU in the REPowerEU initiative under the European Green Deal [5].



**Figure 1:** Map of current electrolyzer projects in Denmark

The value of data is becoming apparent throughout different industries, with the power sector being no exception<sup>2</sup>. By increasing both the amount, the availability, and the quality of data in the power sector, new insights and potentially new methods of operation can be unlocked, which bring the potential of increased efficiency and profit for all market participants. By utilizing data related to the predicted quantities in a

<sup>1</sup>See report from Hydrogen Denmark [7] and the official ptX strategy for the Danish government [4].

<sup>2</sup>Referring to [1] specifically.

---

feature-driven model, better and more stable predictions might be obtained.

This project investigates how feature-driven models can be used to operate an electrolyzer unit in combination with a wind farm, from the perspective of a wind and hydrogen producer acting in the Danish electricity markets and a bilateral hydrogen market. This project is thereby placed primarily within the fields of machine learning, operations research, and decision-making under uncertainty. The target audience is academic researchers and potentially wind farm & ptX-facility operators.

The report builds on the method and findings presented in the paper "Feature-driven Improvement of Renewable Energy Forecasting and Trading" by Muñoz et al. found in [12], where a feature-driven model is used to improve both the accuracy of the forecasted production as well as the expected profit from market participation, from the perspective of a wind producer. The main contribution of this thesis lies in the addition of hydrogen production to this model, and the evaluation and insights of doing so. The development and investigation of new feature-driven models for operation in general, and the concept of real-time adjustment of electrolyzers, constitutes a substantial part of the contribution as well. This leads to the following problem statement:

**How can an electricity-market agent utilize a feature-driven approach to optimize the operation of electrolyzer units in combination with intermittent power production units?**

With the following specific research questions:

- How should the agent account for the electrolyzer operation when placing bids in a forward market?
- How should the agent schedule their hydrogen production based on available data and forward market information?
- How should the agent adjust their schedule based on revealed data in real time?

---

## 2 Literature Review

Due to the intermittent nature of solar and wind, the market operation of such units is complicated by the introduction of uncertainty in the decision-making process. This has motivated a great deal of literature, exploring different ways to handle this uncertainty. [13] approaches the task as a newsvendor problem with the system status<sup>3</sup> modeled as a Bernoulli variable, and formulates the problem as a distributionally robust optimization (DRO). [15] investigates intra-day trading of wind energy, using deterministic algorithms based on point forecasts of the uncertain parameters.

In addition to the stochastic approach in [13] and the deterministic approach in [15], another possibility is using a learning-based approach. The standard procedures entails a machine learning model with parameters trained on historical data, that is used to improve a deterministic point forecast by considering additional features correlated with the realized production. The improved forecast is then used in a subsequent deterministic optimization, constituting the "predict-then-optimize" framework [8]. An alternative to this procedure is the value-oriented learning approach, where a machine learning model is also used for the final objective<sup>4</sup>, and not only for improving a forecast. [11] uses the value-oriented approach to optimize the forecast of net demand from the perspective of a TSO, where the deviation costs are asymmetric, and the objective of minimizing deviations is thus not coincident with the objective of minimizing costs. [6] investigates the value-oriented approach from the perspective of a producer, where two approaches are introduced:

- (1) One model for improving the forecast<sup>5</sup> and a separate model for the value-oriented decision-making, and (2) a single model that features the point forecast directly in the value-oriented decision-making.

Modeling hydrogen production in a ptX context requires that the stochastic production of renewables is addressed when considering the operation of the electrolyzer unit. [17] models the operation of a fleet of electrolyzers in combination with a wind farm, accounting for the operational constraints and costs of the electrolyzers, searching for the optimal number of units to install. [18] investigates a similar situation, where the developed model maximizes the revenue generated from a day-ahead dispatch of an electrolyzer based on a wind forecast, accounting for all the operational constraints as well. In the present project, the possibility of real-time adjustment of the electrolyzer based on realized production is investigated, which is a mechanism that has not been found to be investigated in the reference literature. To this end, significant assumptions are made on the operational constraints of the electrolyzer, which is presented in chapter 4.

The main paper that this project builds upon is [12]. The paper considers the forecasting and trading of wind energy, using the aggregated wind power production in western Denmark as a proxy variable of the power being produced by a wind farm, and the associated forecast issued by Energinet as the available production forecast. A value-oriented approach is used, where a general linear model is trained on historical data to maximize revenue. The training of the parameters is based on the news-vendor formulation, where the cost of deviating from the forward bid is minimized. The paper investigates how including different features in a value-oriented model can improve the resulting revenue, compared to a deterministic model bidding the forecasted value directly. The paper finds that using separate models for improving the forecast and providing the bid<sup>6</sup> results in improved performance.

The present project uses the findings of [12] as a starting point, and expands the research by including the possibility of producing hydrogen. In addition to the investigation of hydrogen production, the present project attempts to improve the evaluation of the results by utilizing more realistic data and additional

---

<sup>3</sup>Whether the grid has a surplus or deficit of electricity.

<sup>4</sup>In this case maximizing revenue of the producer.

<sup>5</sup>By minimizing deviation from realized production based on historical data.

<sup>6</sup>The same procedure investigated in [6].

---

benchmarks. Instead of using the aggregated power as a proxy, actual production forecasts and realizations provided by Siemens Gamesa is used in the evaluation of the developed models. In addition to a deterministic benchmark bidding the forecasted production directly, this project develops several upper bounds on the performance of the models, enabling a more holistic evaluation as well as an estimate for the possible improvement.

---

### 3 Data

Forecast data for both price and production have been provided by Siemens Gamesa (SG). The price forecasts are provided for the period 01/01/2021-30/09/2022, and the production data for the period 01/05/2022-31/10/2022. The price and wind forecasts obviously need to cover the same period to be usable as a dataset, and the common period between the two datasets is 01/05/2022-30/09/2022, corresponding to five months of data. Both price and production data are known to have seasonal variations, which will not be captured by considering just five months of data. Furthermore, since the process requires splitting the data into a training set and a test set, the generality of the results will be reduced even further. In order to establish a general performance of the models across seasonal variations, at least a year of testing data would be required, and an additional year of training data would be preferable to allow a thorough investigation of the length of the training period. To achieve this, synthetic forecasts are created for both price and production data, based on the statistical characteristics of the data provided by SG. By creating synthetic data, a larger dataset can be used for both training and evaluation. Two years of synthetic forecasts will be created in total to construct a year of training data and a year of testing data. The process of generating the forecasts are described in detail in sections 3.1.1 and 3.2.1.

The European Network of Transmission System Operators for Electricity (ENTSO-E), is an association for the cooperation of European Transmission System Operators (TSOs) that are responsible for the operation of Europe's electricity system. They gather data from the electricity grids across europe, and present it on their Transparency Platform<sup>7</sup>. All realized balancing prices and forward prices are exported from here, as well as production realizations of the wind farm Roedsand II which is used as realized production data, and aggregated forecasts issued by the Danish TSO Energinet.

The first of November 2021, the danish TSO Energinet changed the balancing pricing scheme from dual to single. Since only the dual pricing scheme is considered in this project<sup>8</sup>, this change has been ignored, and the data has been used as a dual balancing price, where the forward price is used for balancing settlements in the grid's favor.

All the used data has an hourly time resolution.

#### 3.1 Price Data

This section constructs the price forecast used in the deterministic model, and conducts an in-depth statistical investigation of the forward prices used in the dataset to draw insights about the price data. The price data consists of the realized forward and balancing prices from [2], and the forward price forecast from SG.

##### 3.1.1 Generating Price Forecast

To generate new forecasts, the statistical characteristics of the true forecasts must be investigated. The price forecast  $\hat{\lambda}_t^F$  can be thought of as a noisy version of the realized price  $\lambda_t^F$ :

$$\hat{\lambda}_t^F = \lambda_t^F + \epsilon_t \quad \forall t \in T \tag{1}$$

Where  $\epsilon$  is a random variable. It is readily seen that the value of the noise for each datapoint can be found by calculating the forecasting error as the difference between the forecasted and realized value.

---

<sup>7</sup>The platform can be accessed by [2].

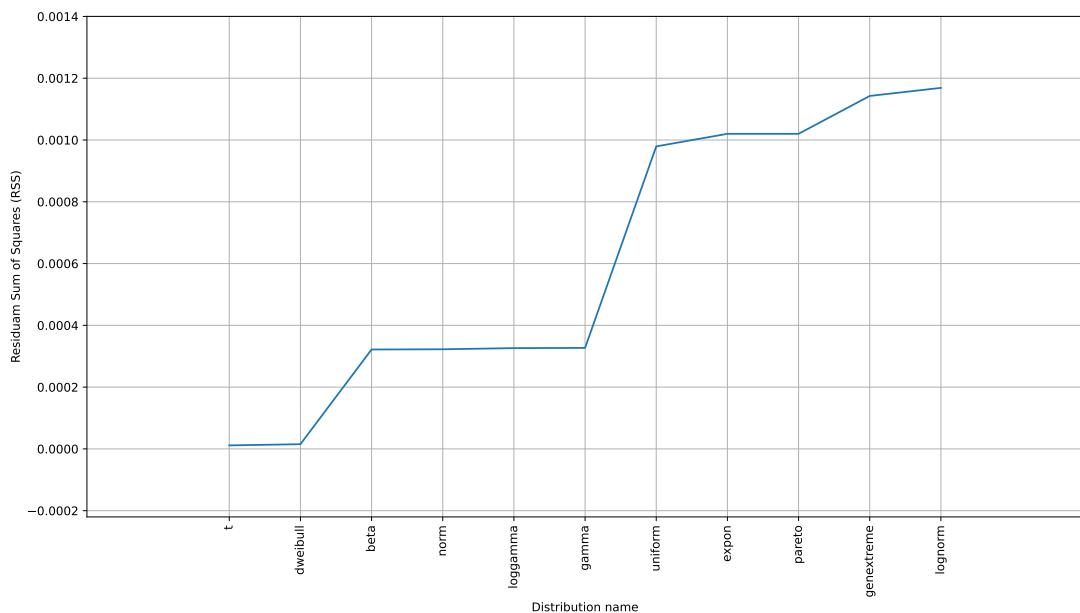
<sup>8</sup>As will be elaborated in section 5.1.1

---


$$\epsilon_t = \hat{\lambda}_t^F - \lambda_t^F \quad \forall t \in T \quad (2)$$

By approximating the distribution of the error values with a theoretical distribution<sup>9</sup>, random values can be drawn from this distribution and added to other realized forward prices to construct new forecasts. This will result in the constructed forecasts displaying the same statistical qualities as the original forecasts.

Figure 2 shows the residual sum of squares (RSS) of a series of distributions fitted to the forecasting error values. The lower the RSS value, the better the fit.

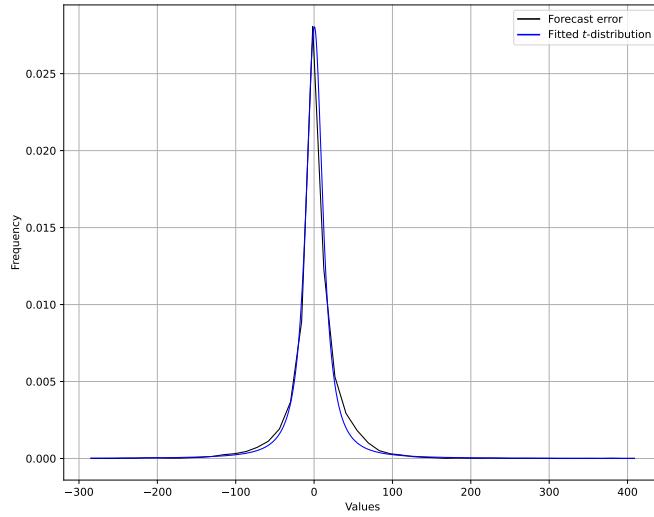


**Figure 2:** Summary of price forecast error distribution fittings

It is seen that the distribution that has the closest fit to the data is a *t*-distribution. The specific parameters and the actual fit of the distribution is plotted on figure 3 along with the forecast errors. Since this is the closest model to the data, and a *t*-distribution is generally an appropriate choice when modelling symmetrically distributed and (somewhat) bell-shaped data with unknown variance, this distribution will be used to draw noise samples for constructing new forecasts.

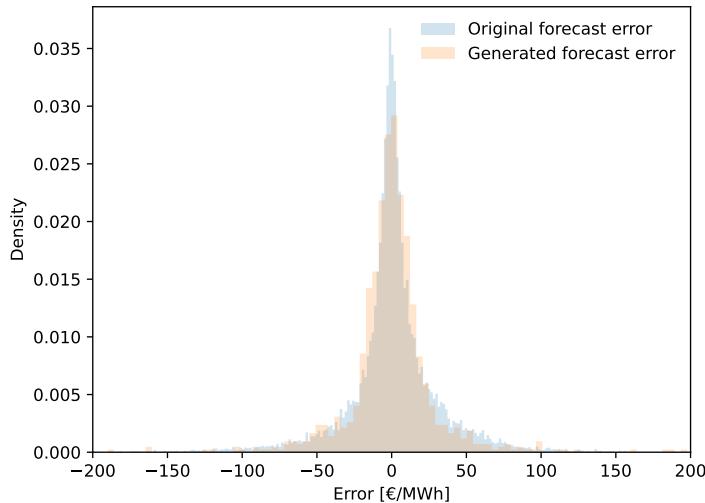
---

<sup>9</sup>The distfit package from [16] has been used to perform the distribution fittings.



**Figure 3:** Price forecast error distribution fitting

Using the found distribution parameters, new noise samples can be drawn. 17.520 samples are drawn, corresponding to two full years with hourly samples. A density plot of the generated samples plotted along with the original values can be seen on figure 4.



**Figure 4:** Synthetic price forecast errors

The noise samples are simply added to the realized forward prices for the period used in the final dataset.

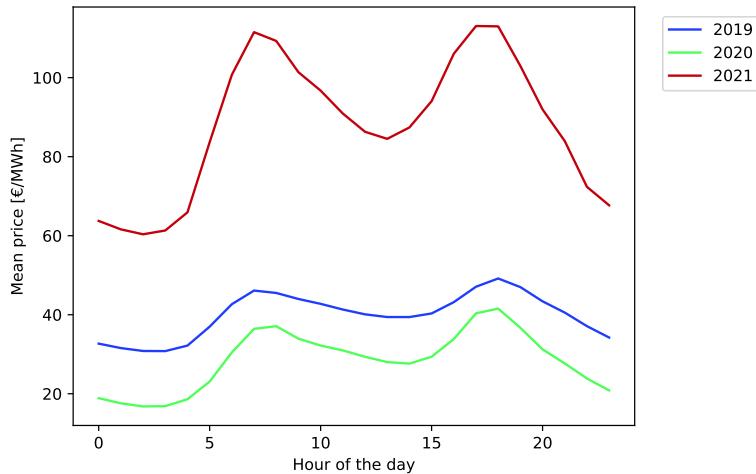
### 3.1.2 Data analysis

The idea of a feature-driven model is to use tunable parameters to capture relational patterns between a set of features, and a set of targets. This is achieved by using historical data to find patterns, which are expected to persist into new data the model is applied on. If there are no such patterns between the features

---

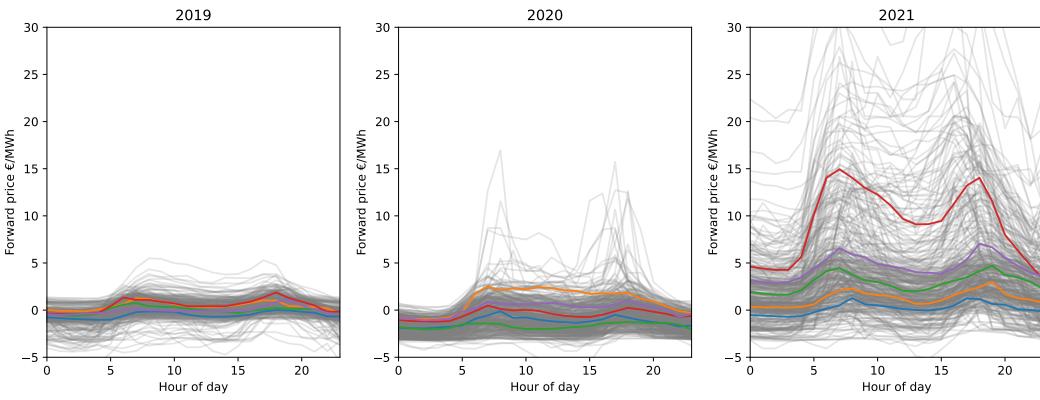
and targets, or the model is not constructed appropriately to find the patterns that exist, the model cannot be expected to produce any usable results. If the patterns *change* between the period of training and the period of application (testing), the same is true. When the patterns and statistics are constant over time, it is said that the model operates in a *stationary* environment, and when they are changing the environment is said to be non-stationary. For completely stationary environments, the longer the training period the better, because the empirical statistics of the dataset will converge to the true statistics. For non-stationary environments, shorter training periods will outperform longer training periods, because the statistics of the most recent data will resemble the statistics of future data more closely.

The arrival of Covid19 had impacts on many aspects of society, electricity prices included. Along with other global events, this caused the electricity price to rise significantly in 2021, compared to 2020. Figure 5 shows what a "mean price day" looks like for each of the years 2019, 2020 and 2021. A "mean price day" is showing the forward prices for each hour of the day, where each hourly value is the mean price in that hour across the year, thus preserving the structure of the daily pattern.



**Figure 5:** Mean price days 2019-2021

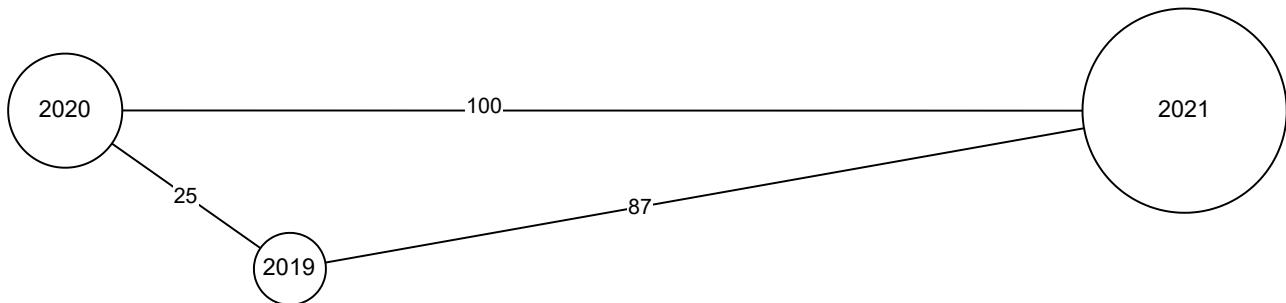
It is clearly seen that the prices has a significant rise in 2021 compared to the other years. To get a more in-depth look into how the prices have changed, figure 6 shows the result of a  $k$ -medoid clustering performed on the price data using  $k = 5$ , where the data has been normalized to 2019 mean and standard deviation. Each gray curve represents a single day of the dataset (called a "price day" from hereon), and the five colored curves are the resulting medoids. The three plots have the same scale on the y-axis, so the data from the different plots is directly comparable. Note that  $k$ -medoids picks the  $k$  most representative price days and thus preserves the volatility of the individual price day, contrary to  $k$ -means clustering which flattens out individual variations. This means the variation in price throughout the day is preserved, and can therefore be evaluated as well.



**Figure 6:** K-medoid clustering of price days - normalized to 2019

It is seen how the prices of 2019 has a small degree of variation, with all the medoids being on top of each other, and small deviations around the dense band of datapoints around the medoids. 2020 shows a larger spread, with significantly more datapoints reaching a very low price, as well as a few large spikes in prices. 2021 clearly shows a much larger spread as well as a general increase, where one of the medoids even shows higher prices than the highest from 2020. The difference in distribution on figure 6 indicates that the forward price is non-stationary throughout the years considered.

One way to quantify the difference in distributions is through a metric called the Wasserstein distance. The Wasserstein distance, also known as the Earth Mover's Distance, is defined as the minimum cost of turning one distribution into another, where the cost is defined as the amount of "mass" that needs to be moved multiplied by the distance it needs to be moved. When considering the distribution of price days, this translates to the number of prices that need to be adjusted, multiplied by the amount they need to be adjusted, for one year to be turned into another. Figure 7 shows a relative measure of similarity between the distributions of price days for each year, calculated based on this metric. The area of the circles indicate the spread of the distribution, and the length of the edges indicate the Wasserstein distance between the distributions. Note that the actual values have no intrinsic meaning with the method applied, and only serves as a relative measure<sup>10</sup>. A detailed description of the calculation and the results can be found in section A.2 in the Appendix.



**Figure 7:** Schematic of price day distributions

The first thing to note is how the spread of the distributions is smallest for 2019, and gets progressively bigger for 2020 and 2021, as was expected by observing figure 6. An interesting result is that the distribution of price days in 2019 is actually more similar to 2021 than to 2020. This is most likely due to 2019 being such

<sup>10</sup>The longest distance of 100 thus allows the remaining distances to be intuitively understood as a percentage relative to the longest.

---

a consistent year, where 2020 had significantly more price days with very low prices compared to both 2019 and 2021.

The key insight from figure 7 is that the forward price shows a significantly larger degree of non-stationarity between 2020 and 2021, than between 2019 and 2020. Since there is non-stationarity of the price data observed from figure ?? between all years, it is expected that increasing the length of the training period will not result in increasing performance indefinitely. At some point, the non-stationarity will cause increasing training periods to result in decreasing performance<sup>11</sup>. Whatever that point will be, it will be at a shorter training period length when testing on 2021, than when testing on 2020, which is worth noting.

The changes in electricity price can to a large degree be explained by various global geopolitical events, meaning: (1) The effects might pass again within a reasonable time frame, and (2) extra caution should be taken when using a model dependent on historical data in times of geopolitical distress. It seems natural that certain types of models are most appropriate in times of greater uncertainty, and other types are most appropriate in times of stability. Longer training periods favor times of stability due to the dependency on historical data, and shorter training periods are preferred when observing short term changes in the predicted quantity. Analyzing the used data in depth is therefore a necessary exercise in order to construct appropriate models, and to draw valid conclusions based on the results. The project will use data from 2019 as training data and 2020 as testing data, and the performance is thus expected to increase as the length of the training period increases for most of the available training set. It should thus be noted as an important insight that the achieved results might be significantly different when applied to highly non-stationary environments.

This analysis could extend even further into the balancing prices, but will be limited to the forward prices, where several important insights have been gathered.

### 3.2 Production Data

The production data provided by SG is a forecast and realization of a 10-minute average wind speed for the period 01/05/2022-31/10/2022. The wind speed is converted into wind power using the following technical parameters:

$$v_{\text{cut-in}} = 3 \text{m/s} \quad (3)$$

$$v_{\text{cut-out}} = 25 \text{m/s} \quad (4)$$

$$v_{\text{rated}} = 13 \text{m/s} \quad (5)$$

$$\bar{P}^F = 10 \text{MW} \quad (6)$$

$v_{\text{cut-in}}$  is the minimum wind speed at which the turbine starts producing.  $v_{\text{cut-out}}$  is the maximum wind speed the turbine can produce at, after which the production will be curtailed to protect the turbine.  $v_{\text{rated}}$  is the wind speed at which the turbine produces its rated power,  $\bar{P}^F$ .

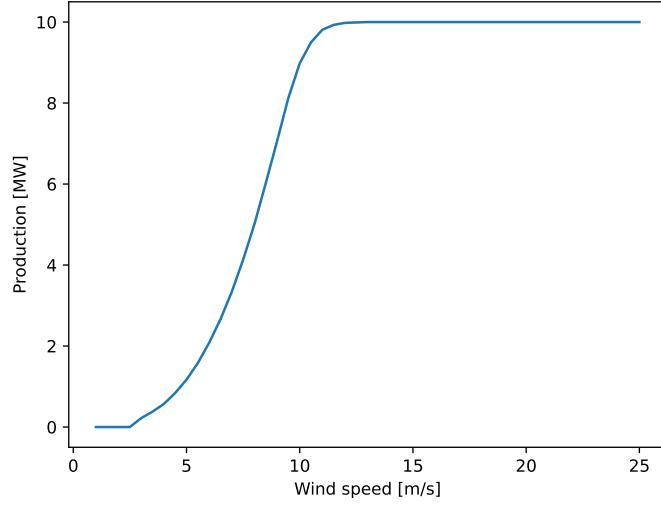
The production at the different wind speeds is divided into four regions, given by the function  $P^W(v)$ :

$$P^W(v) = \begin{cases} 0, & v < v_{\text{cut-in}} \\ 0, & v > v_{\text{cut-out}} \\ \bar{P}^F, & v_{\text{rated}} \leq v \leq v_{\text{cut-out}} \\ E^W(v), & v_{\text{cut-in}} \leq v < v_{\text{rated}} \end{cases} \quad (7)$$

---

<sup>11</sup>Even for completely stationary data the performance increase will stop at some point, but for completely stationary data the performance would plateau and not start deteriorating.

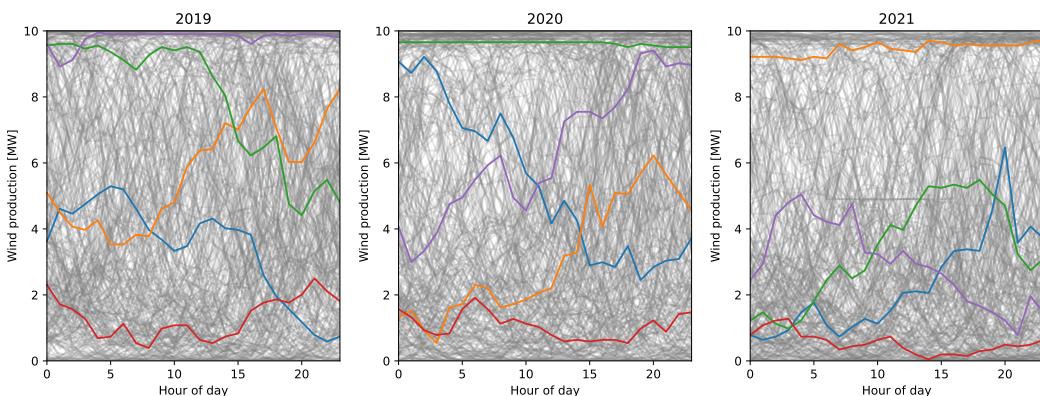
As indicated by definition of the  $P^W(v)$ , the production is varying in the interval  $v_{\text{cut-in}} \leq v < v_{\text{rated}}$  described by the function  $E^W(v)$ . For wind speeds in this interval, the corresponding values of production was included in the data received<sup>12</sup>, with the dependency illustrated in figure 8.



**Figure 8:** Wind power curve

Each datapoint of both forecasted and realized wind speed is converted into production by equation (7), and aggregated to an hourly average, and the data is then ready to be used.

An in-depth analysis of the production data as done with the price data is not performed, because the production data is not dependent on global societal events that can change the nature of the stationarity very quickly, but on the global and local climate which changes much more slowly than a time span of three years. A plot of a  $k$ -medoid clustering of the production data for the three years confirms this expectation, as can be seen on figure 9. This means the production data favors longer training periods than the price data for developing well performing models.

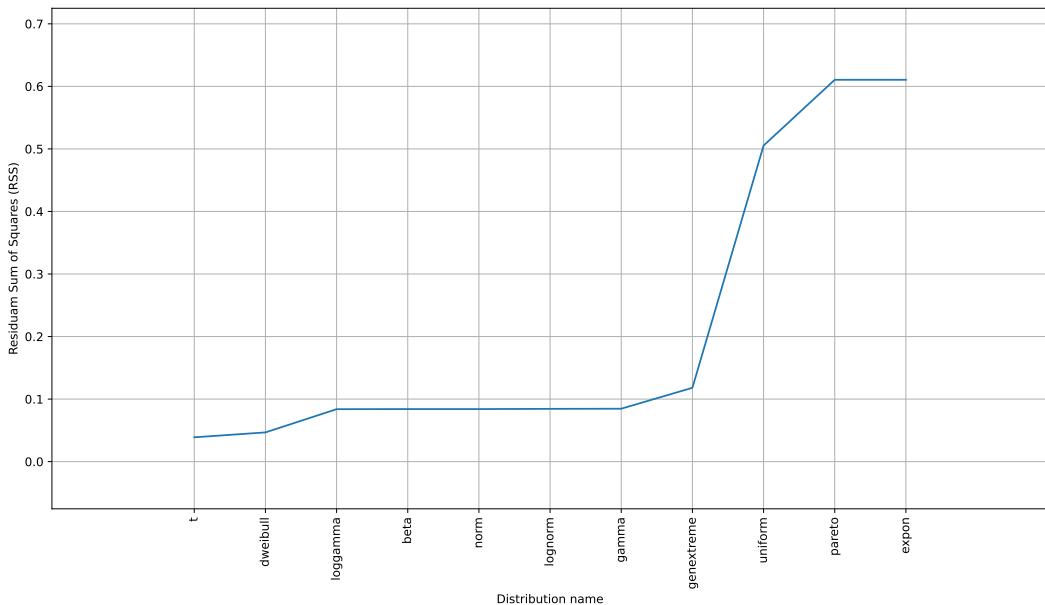


**Figure 9:** K-medoid clustering of production days

<sup>12</sup>The conversion from the wind speed data from SG to wind power values was provided by Bruno Laurini and Alice Patig.

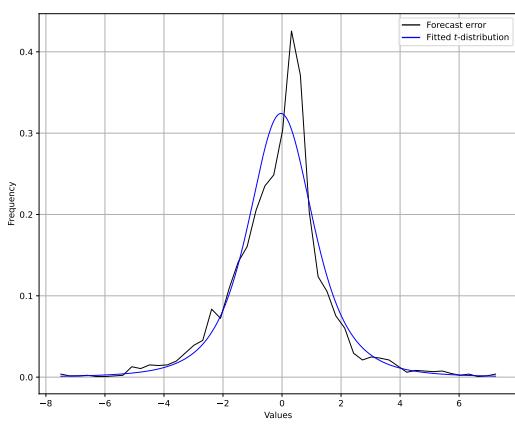
### 3.2.1 Generating Production Forecasts

The production forecasts will be generated following the same procedure as presented in section 3.1.1. A summary of the distribution fittings can be seen on figure 10.

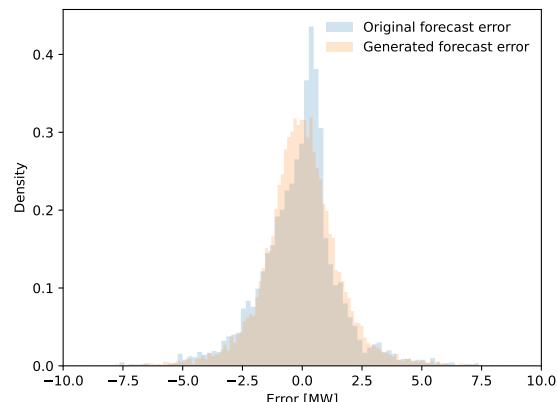


**Figure 10:** Summary of production forecast error distribution fittings

Again, the  $t$ -distribution provides the closest fit to the empirical distribution, which can be seen plotted together with the theoretical fit on figure 11. 17.520 samples are drawn, which can be seen plotted along with the forecasting errors of the actual data on figure 12.



**Figure 11:** Production forecast error distribution fitting



**Figure 12:** Synthetic production forecast errors

The samples are added to the realized production values to construct new forecasts, as was done with the price forecasts.

---

### 3.2.2 Aggregated forecasts

Energinet provides a forecast of the aggregated wind production each hour divided into onshore and offshore for each price area:  $E_t^{Off,DK1}$ ,  $E_t^{Off,DK2}$ ,  $E_t^{On,DK1}$ , and  $E_t^{On,DK2}$ . This forecast is available when the forward bids are to be submitted, and can therefore be utilized as features for the learning-based models<sup>13</sup> when developing forward market bids. This data is extracted from [2].

## 3.3 Summary of Data Chapter

Understanding the data in depth is a task of upmost important when using feature-driven models, as has been demonstrated in subsection 3.1.2. The year 2019 is chosen as training data, and 2020 as testing data, which allows for training periods up to 12 months, and ensures that the models are evaluated throughout the seasonal changes in a year. Forward and balancing prices, as well as aggregated wind forecasts and production realizations are all gathered from [2]. The generated forecasts are applied to the realized productions and forward prices, which completes the final dataset.

---

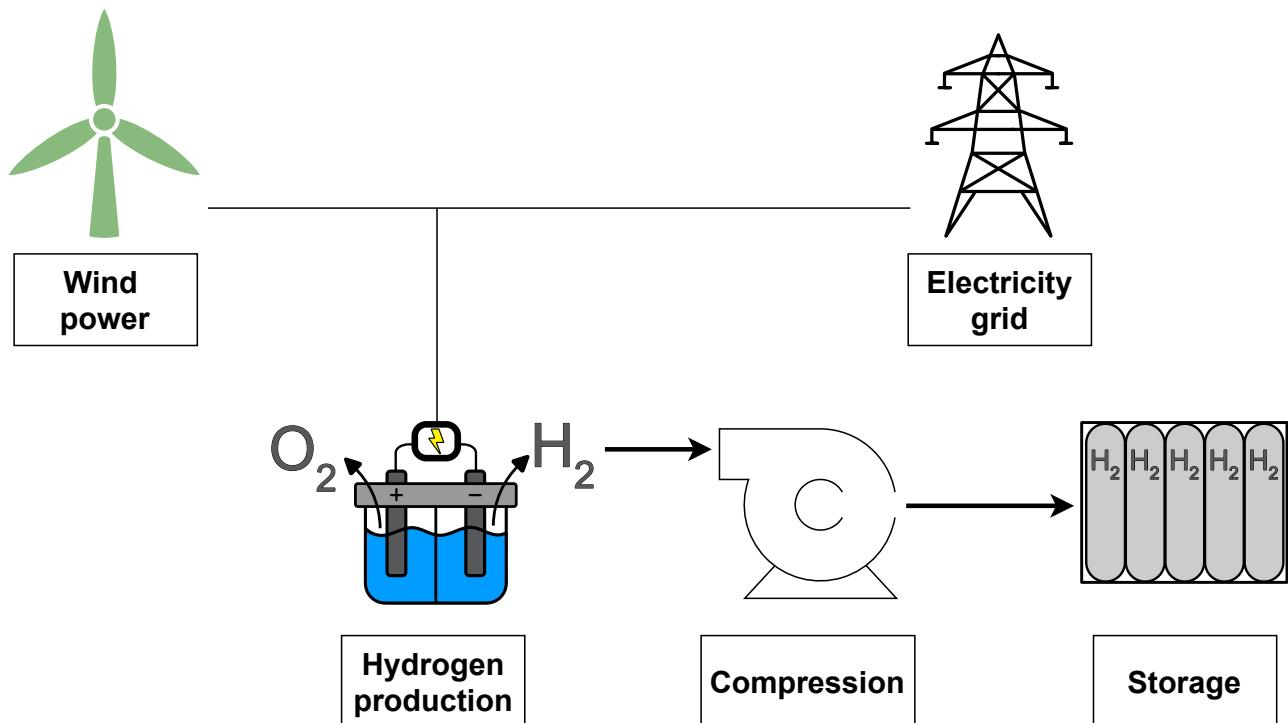
<sup>13</sup>As described in detail in section 5.4.

## 4 Electrolyzer Modelling

This chapter introduces the reader to the electrolyzer technology, and describes the particular modelling choices, assumptions and the main points of simplification.

### 4.1 Electrolyzer technology

Electrolysis is the process of using electricity to split water molecules into oxygen and hydrogen. Generally, electrolysis is considered as a method of producing hydrogen, with oxygen as a byproduct, and therefore only the hydrogen production is considered in this project. A schematic of the main components involved are shown in figure 13.



**Figure 13:** Schematic of hydrogen production

Currently, three different electrolyzer technologies exist:

- Alkaline (AEC)
- Polymer electrolyte membrane (PEM)
- Solid oxide cells (SOEC)

SOEC presents the highest efficiencies, but is in the very early stages of development and not yet mature for industrial deployment<sup>14</sup>. PEM suffers from a dependency on scarce and expensive materials, and combined with a lower efficiency than AEC this results in the technology being generally less attractive<sup>15</sup>. AEC is the most mature technology, with plants on an industrial scale already deployed in a relatively large extent. Combined with a long lifetime, this positions AEC as the first choice for many developers, and the technology is expected to continue constituting the majority of the installed electrolysis capacity across the globe<sup>16</sup>. For this reason, AEC will be used in estimating suitable parameter values throughout the project,

<sup>14</sup>See table 3 p. 100 in [9]

<sup>15</sup>See pp. 101-102 in [9]

<sup>16</sup>See pp. 101-102 in [9]

with a plant size of 10 MW input power.

## 4.2 Conversion and Efficiencies

The efficiency of an electrolyzer is dependent on several different operational parameters, such as output flow and temperature, but will be approximated as constant in this project. This efficiency will account for the complete production trip, including storage, as depicted on figure 13. The production efficiency is given as a conversion factor from [MWh] electricity input to [ $\text{kg}_{\text{H}_2}$ ] hydrogen output, taken directly from the results<sup>17</sup> of section 86 in [9], where this parameter is given as  $\rho_{prod} = 20 \frac{\text{kg}_{\text{H}_2}}{\text{MWh}_{input}}$  for both a 1 MW example plant, and a 100 MW example plant. Storage of hydrogen produces energy losses through valves and tubing as well as standby losses, and since they account for less than 1% of the total energy expenditure in the storage operation<sup>18</sup>, a total value of  $l_{misc} = 1\%$  of the total energy input to the electrolyzer is considered as their contribution. The primary contributor to an effectively lower energy efficiency, is the energy required for compression. The cheapest form of hydrogen storage is tanks made of seamless steel or aluminium, which are pressured to around 200 bar, and are commonly used for stationary applications<sup>19</sup>. Using this type of storage tank, the energy required for compression is approximately<sup>20</sup>  $4 \frac{\text{kWh}}{\text{kg}_{\text{H}_2}}$ . For a given [ $\text{kg}_{\text{H}_2}$ ] of hydrogen with a lower heating value of  $33.33 \frac{\text{kWh}}{\text{kg}_{\text{H}_2}}$ , the final round-trip efficiency for storage can be calculated:

$$\eta_{final} = \frac{\text{Total energy output}}{\text{Total energy input}} - l_{misc} \quad (8)$$

$$\eta_{final} = \frac{33.33 \frac{\text{kWh}}{\text{kg}_{\text{H}_2}}}{33.33 \frac{\text{kWh}}{\text{kg}_{\text{H}_2}} + 4 \frac{\text{kWh}}{\text{kg}_{\text{H}_2}}} - 0.01 \approx 88\% \quad (9)$$

This efficiency is then applied to the initial conversion factor to get the final value conversion efficiency  $\rho_{final}$ :

$$\rho_{final} = \rho_{prod} \cdot \eta_{final} \quad (10)$$

$$\rho_{final} = 20 \frac{\text{kg}_{\text{H}_2}}{\text{MWh}_{input}} \cdot 0.88 = 17.6 \frac{\text{kg}_{\text{H}_2}}{\text{MWh}_{input}} \quad (11)$$

## 4.3 Pricing and Production Quotas

Hydrogen is often sold in bilateral contracts with monthly quotas. The monthly quota is sold at a given price, which might be adjusted based on variable production costs<sup>21</sup>. If the quota is not met, there is a penalty, and if more than the quota is produced, this can be further compensated. The particularities of such a contract, as well as the large inter-temporal constraints caused by the monthly timeframe, is a main point of simplification in the hydrogen modelling used in this project. To take a conservative approach of the length of the contractual time-frame, a daily production requirement is enforced instead, which results in a tighter bound on the model since the space of possible production patterns becomes significantly smaller. It can be a great advantage to locate electrolyzer units not only next to wind parks as considered in this project, but also close to other industrial production facilities which can utilize the hydrogen, thus saving the need for large distance transport. Since the operation of such facilities are outside the scope of this project, selling

<sup>17</sup>See pp. 107-109

<sup>18</sup>See p. 82 in [10]

<sup>19</sup>See table 2 p. 78 in [10]

<sup>20</sup>Value taken from section "Compressor" p. 80 in [10].

<sup>21</sup>An example of such a contract is found in section A.3 in the appendix, where the price is adjusted according to the price of Natural Gas.

---

hydrogen in such a constellation would be modelled using identical conditions. The simplification applied here is thus agnostic of the specific usecase of the hydrogen after production.

The daily production requirement is set at  $880 \frac{\text{kg}_{h2}}{\text{day}}$  corresponding to  $50 \frac{\text{MWh}}{\text{day}}$  of power required for the electrolyzer. With a plant size of 10 MW, this corresponds to 5 full load hours. This is intended to be at a level where producing both more and less than the requirement will be optimal on different days in the dataset, such that a trivial model which is either always producing or never producing will perform very badly. The hydrogen price<sup>22</sup> is assumed constant at  $2 \frac{\text{€}}{\text{kg}_{h2}}$ , and no maximum daily production is implemented. Combined with a final conversion factor  $\rho_{final}$  from section 4.2, this can be expressed as an equivalent price in [€/MWh] of:

$$\lambda^H = \lambda_{kg}^H \cdot \rho_{final} \quad (12)$$

$$\lambda^H = 2 \frac{\text{€}}{\text{kg}_{h2}} \cdot 17.6 \frac{\text{kg}_{h2}}{\text{MWh}_{input}} = 35.2 \frac{\text{€}}{\text{MWh}_{input}} \quad (13)$$

Which is directly comparable to the electricity prices used in the forward market trading and balancing market settlements.

The penalty for not meeting the required daily production is chosen to be  $80.61 \frac{\text{€}}{\text{MWh}} = 4.6 \frac{\text{€}}{\text{kg}_{h2}}$ , corresponding to the 95% quantile of the upwards balancing price for the entire dataset, such that missing the quota results in a loss practically every time, but keeping the results on a realistic level<sup>23</sup>.

#### 4.4 Further Modelling Simplifications

Electrolyzers are often modelled with three possible states of operation: Online, Offline and Standby. Significant start-up costs and ramping constraints are associated with going from the Offline state into Online, and keeping the unit in the standby state has a constant standby cost each timestep<sup>24</sup>. This project does not contain this state modelling, and thereby assumes the electrolyzer is always in standby or Online mode, such that the ramping constraints are on the order of  $25 \frac{\% \text{ of max capacity}}{\text{s}}$ , and thus negligible when considering an hourly time resolution. This means the electrolyzer is assumed capable of real-time adjustment of production without ramping constraints.

---

<sup>22</sup>This value is based on a rounded average of the hydrogen price forecast from Energinet in [3].

<sup>23</sup>The upwards balancing price is used to emulate the necessity of having to buy power at the last minute to meet the daily production requirement.

<sup>24</sup>See both [17] and [18] for examples of such a model formulation

## 5 Model Development

This chapter begins by developing the mathematical formulation of wind power trading in a forward market with subsequent settlements of deviations, and then continues to augment that formulation with the addition of hydrogen production and the associated constraints. A deterministic model and several feature-driven models with their associated algorithms will then be developed based on this formulation.

Table 1 shows an overview of the constants used in the present chapter.

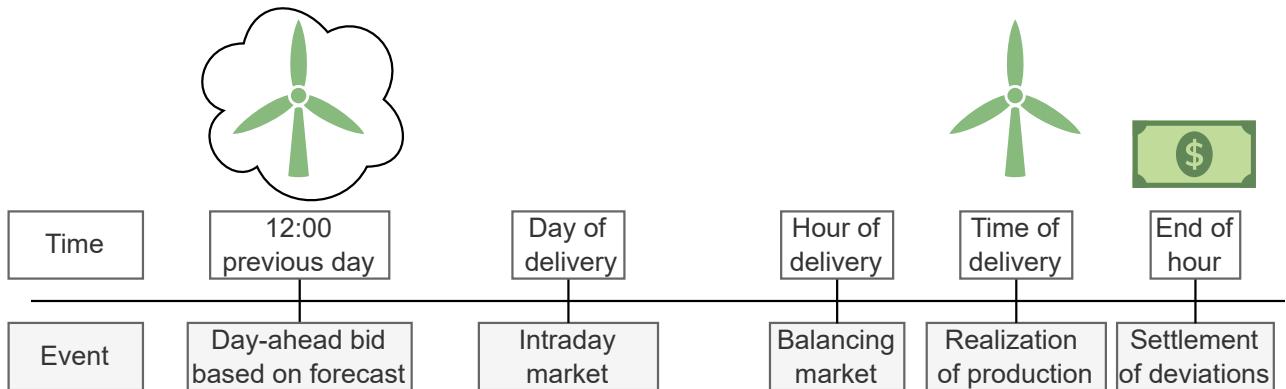
Name	Description	Value
$\bar{P}^F$	Rated wind power	10.00[MW]
$\bar{P}^H$	Rated electrolyzer power	10.00[MW]
$P^{DMin}$	Daily $H_2$ MWh-equivalent production requirement <sup>25</sup>	50.00[MWh/day]
$\psi$	Penalty for missing production <sup>25</sup>	80.61[€/MWh]
$\lambda^H$	Hydrogen price <sup>25</sup>	35.20[€/MWh]

**Table 1:** Overview of constants

### 5.1 Mathematical Formulation

#### 5.1.1 Trading Wind

A general timeline of the different markets faced by a wind producer is illustrated on figure 14.



**Figure 14:** Timeline of electricity markets

When the producer wants to sell their production, they must do it the previous day in the forward market<sup>26</sup>, even though the production at this time is uncertain. Production in each hour of the following day is committed the day before in this market. Up until the hour of delivery, the intra-day market can be used to trade power with other market participants, in case the producer receives an updated forecast that differs from the one used in the forward market. The balancing market is used by the TSO to secure any generation or consumption required for the stability of the grid. At the time of delivery, the actual production is realized,

<sup>25</sup>Developed in section 4.3

<sup>26</sup>The description here matches the Day-ahead market in Nordpool, but the more general expression "forward market" will be used throughout the report, since the developed methods does not depend on this specific time-frame.

---

and any deviations are settled afterwards, where the producer is compensated the down-regulation balancing price for surplus production, and charged the up-regulation balancing price for deficit<sup>27</sup>. This report considers only the initial bid in the forward market, and the settlement of deviations from this bid after realization.

The objective of a wind producer is thus to maximize their revenue from the combined participation in the forward market, and the settlement of deviations afterwards. The initial bid in the forward market  $p_t^F$  is compensated in the by the price  $\lambda_t^F$ , and deviations from this bid are settled after the production is realized with the prices  $\lambda^{DW}$  and  $\lambda^{UP}$ , for surplus and deficit production respectively. The total revenue thus consists of the revenue from the forward market  $\lambda_t^F p_t^F$ , the revenue from realized surplus production  $\lambda^{DW}(E_t^{real} - p_t^F)^+$ , and the penalty from realized deficit production  $\lambda^{UP}(p_t^F - E_t^{real})^+$ , where the  $(\cdot)^+$  operator is shorthand for  $\max(0, \cdot)$ . The total revenue for any given hour can thus be written as the function  $\mathbf{R}^{org,dual}$ , which takes a series of forward bids  $\mathbf{p}^F$  as input:

$$\mathbf{R}^{org,dual}(\mathbf{p}^F) = \sum_{t \in T} \left[ \lambda_t^F p_t^F + \lambda^{DW}(E_t^{real} - p_t^F)^+ - \lambda^{UP}(p_t^F - E_t^{real})^+ \right] \quad (14)$$

The function describing the total revenue obtained by the outputs of a model will from hereon be referred to as *The revenue function*, which is not necessarily equal to the objective function of a model. The revenue function serves as a common metric of evaluation between the different models and algorithms they appear in, and describes the contextual framework the model is evaluated in, rather than anything specific to the model. Note that since the revenue function is used for evaluating a model *after* it has been applied on a dataset, there is no requirements for convexity. In the present model formulation, equation (14) both describes the relevant revenue function, and the objective of the model being developed, but to use it as objective function, the  $(\cdot)^+$  operator will need to be addressed at some point.

The formulated revenue function holds true for the dual pricing scheme, where producing more than the initial bid results in compensation less than or equal to what was gained in the forward market, given by the price rule:

$$\lambda_t^{DW} \leq \lambda_t^F \quad \forall t \in T \quad (15)$$

Producing less than the initial bid effectively results in the producer having to buy the missing production, at a price greater than or equal to the forward price:

$$\lambda_t^{UP} \geq \lambda_t^F \quad \forall t \in T \quad (16)$$

The actual prices  $\lambda^{DW}$  and  $\lambda^{UP}$  depends on the status of the grid, which can be in either a state of surplus or deficit production, or calibrated within its margin of tolerance. A local surplus in a grid deficit, or vice versa, results in no penalty, since the producer is effectively balancing the grid. A local surplus in a grid surplus, or vice versa, results in a non-zero penalty, since the producer is amplifying the grid imbalance. Deviation in a calibrated grid is not penalized, meaning both  $\lambda^{UP}$  and  $\lambda^{DW}$  are equal to  $\lambda^F$ . Since the grid can only be either in surplus or deficit or calibrated, at least one of the balancing prices must be equal to the forward price, as described by equation (17).

$$(\lambda_t^{UP} - \lambda_t^F)(\lambda_t^{DW} - \lambda_t^F) = 0 \quad \forall t \in T \quad (17)$$

---

<sup>27</sup>Assuming a dual-price balancing scheme is used, which will be further elaborated later in the section.

---

In a single pricing scheme, both directions of local deviations are settled with the same price, meaning local deviations that help the grid restore balance is rewarded with a net gain in revenue, and only deviations worsening the grid status is penalized. This simplifies the revenue function as such:

$$\mathbf{R}^{org,sngl}(\mathbf{p}^F) = \sum_{t \in T} \left[ \lambda_t^F p_t^F + \lambda^B (E_t^{real} - p_t^F) \right] \quad (18)$$

However, this complicates the operation strategy, since it introduces the possibility of arbitraging between the forward and balancing market and profiting from deviations. This would shift the focus towards a prediction of the future system status, which is not the intention of this report, and the dual pricing scheme is therefore used throughout the developed models.

The model formulation for a wind farm participating in the electricity market as described consists of maximizing the revenue function for all timesteps. Since no consumption is possible for the wind farm, the forward bid is constrained to be between 0 and maximum production  $\bar{P}^F$ , because any bid outside this interval is guaranteed to result in a real time deviation which is by definition sub-optimal.

$$0 \leq p_t^F \leq \bar{P}^F, \quad \forall t \in T \quad (19)$$

The  $(\cdot)^+$  operator is handled by introducing a binary variable  $b_t$ , which will take the value 1 if there is a deficit after the realization, and 0 if there is a surplus. If there is no deviation, the value will have no effect since a surplus of 0 is equivalent to a deficit of 0, and in that case it will be decided by numerical truncation in the solver. Along with the binary variable, three helper variables are introduced:  $E_t^{settled}$ ,  $E_t^{DW}$  and  $E_t^{UP}$ , which are explained as they appear in the formulation.  $E_t^{settled}$  could technically be excluded, but functions as an explanatory and notational convenience.

The net amount of energy settled in the balancing stage is held by the free variable  $E_t^{settled}$ , which consist of the term inside the first operator  $(\cdot)^+$ , and will thus be positive for surplus production, and negative for deficit:

$$E_t^{settled} = E_t^{real} - p_t^F \quad \forall t \in T \quad (20)$$

The non-negative variable  $E_t^{DW}$  holds the surplus deviation, and should take the value of  $E_t^{settled}$  if this is positive, or else it should be equal so 0. The lower bound on  $E_t^{DW}$  is thus fully defined by the following two constraints:

$$E_t^{DW} \geq 0 \quad \forall t \in T \quad (21)$$

$$E_t^{DW} \geq E_t^{settled} \quad \forall t \in T \quad (22)$$

Since a surplus deviation appears as a positive term in the maximized objective function (14), an upper bound on the variable is required for the model to be bounded. This is achieved by using big- $M$  notation, where a large number  $M$  is introduced along with two complementary constraints:

$$E_t^{DW} \leq E_t^{settled} + M \cdot b_t \quad \forall t \in T \quad (23)$$

$$E_t^{DW} \leq M \cdot (1 - b_t) \quad \forall t \in T \quad (24)$$

If  $b_t = 1$ , constraint (24) will force  $E_t^{DW}$  to zero, and constraint (23) will not be binding. If  $b_t = 0$ , constraint (23) will upper bound  $E_t^{DW}$  to  $E_t^{settled}$ , and constraint (24) will not be binding. Note that this requires  $M$  to be large enough that the following relation will always hold:

---


$$E_t^{settled} + M \geq 0 \quad \forall t \in T \quad (25)$$

By equation (20), this requires that the forward bid  $p_t^F$  is upper bounded. The forward bid is not necessarily bounded by the size of the wind farm. Indeed, speculating between the forward market and balancing stage by offering more than the expected production could be a source of profit when performing the arbitrage that can be leveraged in a single pricing scheme, and it is worth noting that this point could be addressed if the work of this project is adapted to a single pricing scheme in the future. In the present model though, any bid larger than the maximum wind farm capacity is guaranteed to result in a deviation, and no profit can be gained from deviating. So as long as  $M \geq \bar{P}^F$ , equation (25) will hold even for the extreme case of  $E_t^{real} = 0$ . Since  $M$  is a constant and not a variable, this is not included as a constraint in the model formulation.

The case is similar for  $E_t^{UP}$  holding the deficit, where the case of  $E_t^{settled} < 0$  is handled instead, meaning the sign in front and the cases for  $b_t$  are therefore switched:

$$E_t^{UP} \geq 0 \quad \forall t \in T \quad (26)$$

$$E_t^{UP} \geq -E_t^{settled} \quad \forall t \in T \quad (27)$$

$$E_t^{UP} \leq -E_t^{settled} + M \cdot (1 - b_t) \quad \forall t \in T \quad (28)$$

$$E_t^{UP} \leq M \cdot b_t \quad \forall t \in T \quad (29)$$

Constraints (26) and (27) provide the lower bound on  $E_t^{UP}$ . To reduce the solution space, (28) and (29) are included as well, even though  $E_t^{UP}$  appears in a negative term in equation (14), and would therefore be minimized by the solver.

This results in the following model definition for the problem of a wind producer:

$$\underset{\mathbf{p}^F}{\text{maximize}} \quad \sum_{t \in T} [\lambda_t^F p_t^F + \lambda_t^{DW} E_t^{DW} - \lambda_t^{UP} E_t^{UP}] \quad (30)$$

s.t.

$$p_t^F \leq \bar{P}^F \quad \forall t \in T \quad (31)$$

$$p_t^F \geq 0 \quad \forall t \in T \quad (32)$$

$$E_t^{settled} = E_t^{real} - p_t^F \quad \forall t \in T \quad (33)$$

$$E_t^{DW} \geq E_t^{settled} \quad \forall t \in T \quad (34)$$

$$E_t^{DW} \leq E_t^{settled} + M \cdot b_t \quad \forall t \in T \quad (35)$$

$$E_t^{DW} \leq M \cdot (1 - b_t) \quad \forall t \in T \quad (36)$$

$$E_t^{UP} \geq -E_t^{settled} \quad \forall t \in T \quad (37)$$

$$E_t^{UP} \leq -E_t^{settled} + M \cdot (1 - b_t) \quad \forall t \in T \quad (38)$$

$$E_t^{UP} \leq M \cdot b_t \quad \forall t \in T \quad (39)$$

$$p_t^F, E_t^{UP}, E_t^{DW} \in \mathbb{R}^+ \quad \forall t \in T \quad (40)$$

$$E_t^{settled} \in \mathbb{R} \quad \forall t \in T \quad (41)$$

Where  $\mathbf{p}^F$  is the vector whose elements are  $p_t^F$ .

### 5.1.2 Adding Hydrogen

As developed in section 4.3, the hydrogen price  $\lambda^H$  is considered constant, and the conversion efficiency is contained in the price to ease notation. The revenue from hydrogen production is thus given by the

---

production of hydrogen  $p_t^H$  as  $\lambda^H p_t^H$ . The power directed to the electrolyzer must be considered in the real time production settlement as well, and should thus be included in the balancing terms inside the  $(\cdot)^+$  operators. The addition of hydrogen comprises a new contextual framework, and thus an updated revenue function:

$$\mathbf{R}^{upt}(\mathbf{p}^F, \mathbf{p}^H) = \sum_{t \in T} \left[ \lambda_t^F p_t^F + \lambda_t^H p_t^H + \lambda_t^{DW} (E_t^{real} - p_t^F - p_t^H)^+ - \lambda_t^{UP} (p_t^F + p_t^H - E_t^{real})^+ \right] \quad (42)$$

$$\mathbf{R} = \sum_{t \in T} \left[ \lambda_t^F p_t^F + \lambda_t^H p_t^H + \lambda_t^{DW} (E_t^{real} - p_t^F - p_t^H)^+ - \lambda_t^{UP} (p_t^F + p_t^H - E_t^{real})^+ \right] \quad (43)$$

The electrolyzer is constrained by its maximum capacity  $\bar{P}^H$ , and is not considered to have the potential to act as a fuel cell, and  $p_t^H$  must therefore be non-negative:

$$0 \leq p_t^H \leq \bar{P}^H, \quad \forall t \in T \quad (44)$$

The minimum production requirement is imposed on a daily basis, where  $T^D$  denotes an appropriate set of indices of the time period corresponding to a day starting at midnight. In a series of timesteps  $T$ ,  $T^D$  will be the notation of a set of indices corresponding to the timesteps of each new day. If  $T$  for example is the hourly timesteps for 3 days (72 hours),  $T^D$  will be defined as such:

$$T = [1..72] \quad (45)$$

$$T^D = \{1, 25, 49\} \quad (46)$$

Where  $[a..b]$  denotes a closed interval of integers.  $T^D$  will be used throughout the report when iterating over a day, where each element  $t^D \in T^D$  corresponds to the first hour in a day, and  $t^D + 23$  to the last hour.

The production requirement is denoted  $P^{DMin}$ :

$$\sum_{t=t^D}^{t^D+23} p_t^H \geq P^{DMin}, \quad \forall t^D \in T^D \quad (47)$$

Since the electrolyzer is a consumption unit, the forward market bid should no longer be constrained to be non-negative, since a negative forward bid (buying power in the forward market) could be result in a net revenue gain in hours with low prices. The forward bid should therefore be lower bounded by the maximum possible consumption, corresponding to the electrolyzer capacity:

$$-\bar{P}^H \leq p_t^F \leq \bar{P}^F, \quad \forall t \in T \quad (48)$$

With these changes, the complete model formulation including hydrogen is given as:

---


$$\underset{\mathbf{p}^F, \mathbf{p}^H}{\text{maximize}} \quad \sum_{t \in T} [\lambda_t^F p_t^F + \lambda^H p_t^H + \lambda_t^{DW} E_t^{DW} - \lambda_t^{UP} E_t^{UP}] \quad (49)$$

s.t.

$$p_t^F \leq \bar{P}^F \quad \forall t \in T \quad (50)$$

$$p_t^F \geq -\bar{P}^H \quad \forall t \in T \quad (51)$$

$$p_t^H \leq \bar{P}^H \quad \forall t \in T \quad (52)$$

$$\sum_{t=t^D}^{t^D+23} p_t^H \geq P^{DMin} \quad \forall t^D \in T^D \quad (53)$$

$$E_t^{settled} = E_t^{real} - p_t^F - p_t^H \quad \forall t \in T \quad (54)$$

$$E_t^{DW} \geq E_t^{settled} \quad \forall t \in T \quad (55)$$

$$E_t^{DW} \leq E_t^{settled} + M \cdot b_t \quad \forall t \in T \quad (56)$$

$$E_t^{DW} \leq M \cdot (1 - b_t) \quad \forall t \in T \quad (57)$$

$$E_t^{UP} \geq -E_t^{settled} \quad \forall t \in T \quad (58)$$

$$E_t^{UP} \leq -E_t^{settled} + M \cdot (1 - b_t) \quad \forall t \in T \quad (59)$$

$$E_t^{UP} \leq M \cdot b_t \quad \forall t \in T \quad (60)$$

$$p_t^H, E_t^{UP}, E_t^{DW} \in \mathbb{R}^+ \quad \forall t \in T \quad (61)$$

$$p_t^F, E_t^{settled} \in \mathbb{R} \quad \forall t \in T \quad (62)$$

This concludes the addition of hydrogen in the mathematical formulation, and model (49)-(62) will be denoted as the **Base model** from hereon.

Note that the requirement on  $M$  now needs to account for the upper bound on the hydrogen production as well, which is given by the size of the electrolyzer unit. This results in the requirement that:  $M \geq \bar{P}^F + \bar{P}^H$ .

Due to the nature of the feature-driven models that will be introduced in section 5.3, constraint (53) can not be guaranteed outside the training environment without applying a specific policy to enforce the production requirement. Instead of applying such a policy, a penalty is imposed on missing production in the revenue function used for evaluating the algorithms. The value of the penalty was argued in section 4.3. The missing production is denoted  $p^M$ , and is the total of the production missing each day (if any), summed over all days considered. The value is calculated by algorithm 1, which requires all the hydrogen production values  $\mathbf{p}^H$  for the considered time period indexed daily by  $T^D$ .  $p^M$  is the running total of missing production for the entire period. Each day,  $p^M$  is updated to include to positive part of the difference between the minimum daily production  $P^{DMin}$  and the sum of the hydrogen produced each hour in that day  $\sum_{t=t^D}^{t^D+23} p_t^H$ .

---

#### Algorithm 1 Missing production

```

Require:  $\mathbf{p}^H$ 
 $p^M = 0$ 
for each day  $t^D \in T^D$  do
     $p^M \leftarrow p^M + \left( P^{DMin} - \sum_{t=t^D}^{t^D+23} p_t^H \right)^+$ 
end for
return  $p^M$ 

```

---

Where the notation  $a \leftarrow a + b$  denotes that the value of  $a$  is updated to be the previous value of  $a$  plus the update  $b$ .

---

The total missing production for the entire time period is then translated to lost revenue by the penalty  $\psi$ , as the term  $p^M\psi$  appearing outside any summation over time periods. This completes the additions resulting from introducing hydrogen, and the *Final revenue function*  $\mathbf{R}(\mathbf{p}^F, \mathbf{p}^H)$  can thus be defined, which will be used to evaluate the performance of the algorithms developed from hereon against each other:

$$\mathbf{R}(\mathbf{p}^F, \mathbf{p}^H) = \sum_{t \in T} \left[ \lambda_t^F p_t^F + \lambda_t^H p_t^H + \lambda_t^{DW} (E_t^{real} - p_t^F - p_t^H)^+ - \lambda_t^{UP} (p_t^F + p_t^H - E_t^{real})^+ \right] - p^M\psi \quad (63)$$

### 5.1.3 Solving the Model

The production  $E_t^{real}$  and prices  $\lambda_t^F$ ,  $\lambda_t^{UP}$  &  $\lambda_t^{DW}$  in the base model (49)-(62) are unknown at the time of bidding. Different methods exist for handling this uncertainty, with the three general approaches being illustrated in figure 15. Blue circles in the figure represent an optimization problem being solved and green filled squares represent data input.

A common characteristic is that some kind of forecast is used. The forecast can be deterministic, where a single value is generated, or probabilistic, where the forecast is characterized by a distribution.

The deterministic approach is the simplest. The procedure here is to ignore the uncertainty, and solve the base model without considering the uncertainty dependent parameters such as real time deviations. Deterministic forecasts are used for the required uncertain parameters such as forward price and production as if they were the realized values. Each time step, a new forecast is generated based on previous forecasts and the applied forecasting model<sup>28</sup>. The deterministic approach will be used as a benchmark in the project, to compare against the models resulting from the learning-based approach.

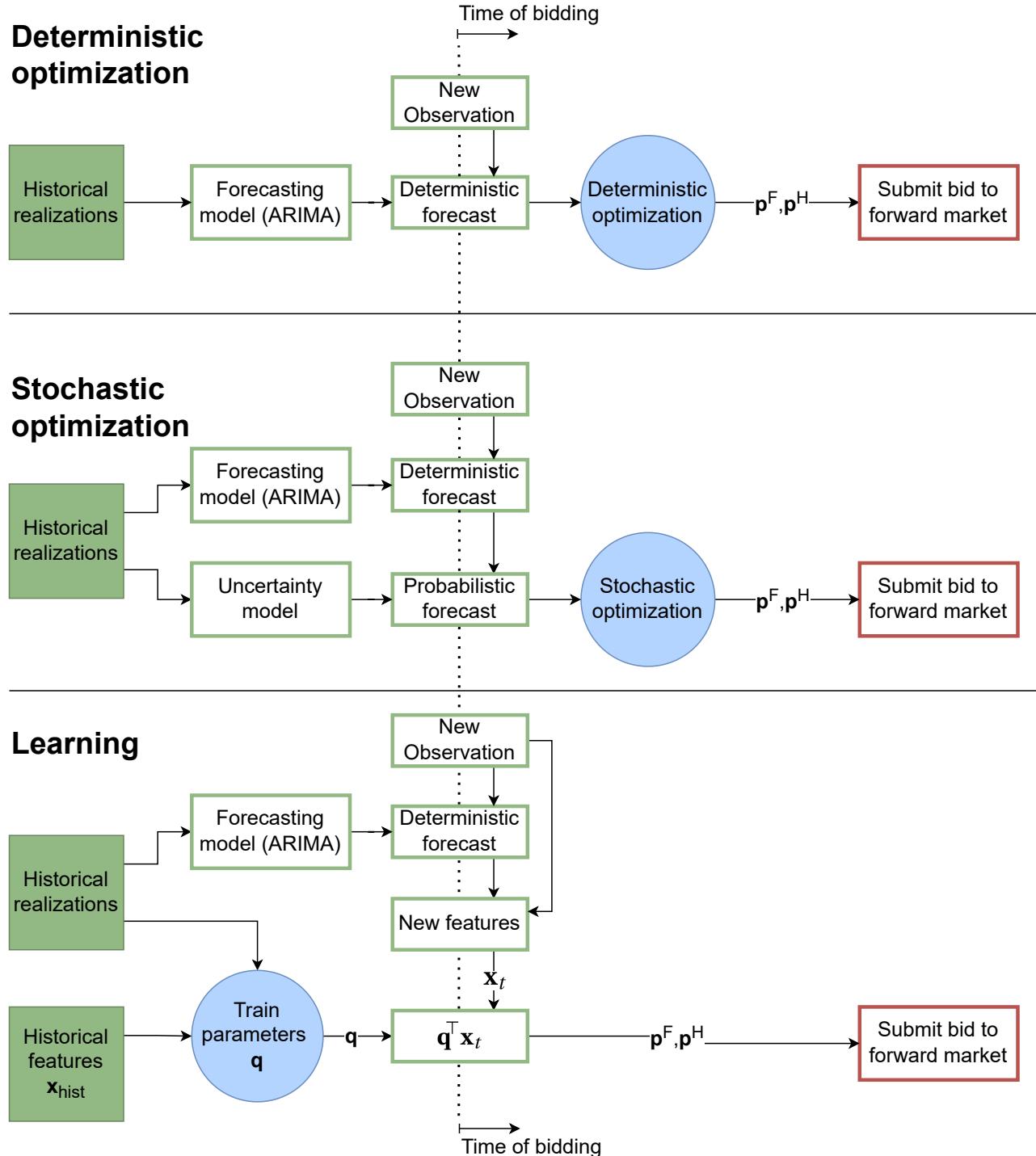
In the stochastic approach, three main methods exist for handling this type of stochasticity in the data: Robust optimization (RO), stochastic programming (SP), and distributionally robust optimization (DRO). RO uses a deterministic forecast, and applies a degree of uncertainty learned from historical data to generate an uncertainty set. The worst-case realizations of these uncertainty sets are used in the optimization problem, thus making the resulting decisions robust<sup>29</sup> to the uncertainty. SP uses a probabilistic forecast, where the distribution of uncertainty is quantified using different realizations of the uncertain parameters weighted by their associated probability, called scenarios. The optimization problem is modified to do a probability weighted sum over the scenarios, where the decision variables are applied identically across all scenarios, thus exposing the model to the statistical information contained in the scenarios. DRO seeks to be robust against the quantification of uncertainty, meaning the distribution of the probabilistic forecast itself is considered uncertain. By considering a family of potential distributions to describe the data, called an ambiguity set, the model seeks to be robust in the case where the true distribution of the data is different than what is modelled in the probabilistic forecast.

The learning-based approach utilizes historical realizations in combination with historical features  $\mathbf{x}_{hist}$ . The features are a set of values that are expected to have some kind of correlation with the realizations, and are available at the time of bidding. A deterministic forecast can thus be included as a feature, along with additional data that might further improve the performance. In a traditional learning-based approach, the intent is to train a series of parameters  $\mathbf{q}$ , that will model the relation between the features and the realizations such that when given a new set of features as input, the model predicts the future realization as accurately as possible. In a value-oriented approach, the intent is not to predict a realization as accurately as possible, but to output a decision variable that optimizes a specific context. As indicated by the figure, the

<sup>28</sup>An ARIMA model is mentioned as an example in the figure.

<sup>29</sup>Robust meaning that the possibility of incurring an unexpected loss is minimized or even eliminated.

training of the model is an optimization problem, and the application is a simple vector multiplication. The obvious benefit of this approach is that the optimization is done based on historical data, and no optimization is required at the time of bidding. Since the optimization constitutes the bulk of the computational effort in all the approaches, deterministic and stochastic methods suffer the consequence that they might not be able to utilize the most recent forecast available simply due to the time required for computation.



**Figure 15:** Different approaches to solving the base model

---

This project investigates the introduction of hydrogen production into the problem of a wind producer, using the learning-based approach and comparing it to the deterministic, leaving the stochastic for future work. The specific type of learning-based model investigated is a value-oriented model. This means that the output of the model is the actual forward market bid, optimized for the objective of the producer, instead of an improved production forecast which would then be used in a deterministic model<sup>30</sup>. The models resulting from the learning-based approach will be denoted as feature-driven models.

## 5.2 The Deterministic Approach

In a deterministic framework, the concept of uncertainty is undefined. The deterministic model must therefore consider the forecast to be the actual realized value, in order to be defined as deterministic. Deviations, and consequently the balancing settlements, should therefore not be a part of the deterministic formulation, and the deterministic formulation will then be a simplified version of the base model.

The model requires a forecast of both production and forward price. The forecast used by the deterministic model will be a set of tuples with price and production forecast for each hour of the day

$$\{(\hat{\lambda}_t^F, \hat{p}_t^{FC}) \mid \forall t \in [t^D..t^D + 23], \forall t^D \in T^D\} \quad (64)$$

The deterministic model will be solved for one day at a time, meaning the set of time steps solved by the model is  $[t^D..t^D + 23]$ , with the model being solved anew for each  $t^D \in T^D$ . The objective of such a model is to maximize the total revenue generated from participating in both the forward market and selling hydrogen for a single day at a time, as expressed in equation (65):

$$\underset{\mathbf{p}^F, \mathbf{p}^H}{\text{maximize}} \sum_{t=t^D}^{t^D+23} [\hat{\lambda}_t^F p_t^F + \lambda_t^H p_t^H] \quad (65)$$

Since the balancing settlements and  $(\cdot)^+$  operator is not present, constraints (54)-(60) from the base model are removed, and only the capacity constraints remain:

$$-\bar{P}^H \leq p_t^F \leq \bar{P}^F \quad \forall t \in [t^D..t^D + 23] \quad (66)$$

$$0 \leq p_t^H \leq \bar{P}^H \quad \forall t \in [t^D..t^D + 23] \quad (67)$$

$$\sum_{t=t^D}^{t^D+23} p_t^H \geq P^{DMin} \quad (68)$$

But an additional constraint arises to enforce the deterministic nature of the production forecast. Deviation is undefined from the deterministic perspective, and the total power allocated to forward market and electrolyzer must therefore equal the total power available:

$$p_t^F + p_t^H = \hat{p}_t^{FC} \quad \forall t \in [t^D..t^D + 23] \quad (69)$$

Note that equation (69) also enforces that any consumption larger than the forecast requires the power to be bought in the forward market.

---

<sup>30</sup>This distinction is highlighted because improving the forecast used in a deterministic model is the most widely applied procedure of implementing learning-based models in wind trading.

---

This results in the following complete model for the deterministic approach:

$$\underset{\mathbf{p}^F, \mathbf{p}^H}{\text{maximize}} \sum_{t=t^D}^{t^D+23} [\hat{\lambda}_t^F p_t^F + \lambda^H p_t^H] \quad (70)$$

s.t.

$$p_t^F \leq \bar{P}^F \quad \forall t \in [t^D..t^D+23] \quad (71)$$

$$p_t^F \geq -\bar{P}^H \quad \forall t \in [t^D..t^D+23] \quad (72)$$

$$p_t^H \leq \bar{P}^H \quad \forall t \in [t^D..t^D+23] \quad (73)$$

$$p_t^F + p_t^H = \hat{p}_t^{FC} \quad \forall t \in [t^D..t^D+23] \quad (74)$$

$$\sum_{t=t^D}^{t^D+23} p_t^H \geq P^{DMin} \quad (75)$$

$$p_t^F \in \mathbb{R} \quad \forall t \in [t^D..t^D+23] \quad (76)$$

$$p_t^H \in \mathbb{R}^+ \quad \forall t \in [t^D..t^D+23] \quad (77)$$

The model is operated by algorithm 2. Each day a set of 24 forecasts is received, the model is solved, and bids are submitted at the forecasted price  $\pm 5\text{€}/\text{MW}$ . If  $p_t^F$  is positive, a production bid is submitted at  $5\text{€}/\text{MW}$  lower than the forecast, and if  $p_t^F$  is negative, a consumption bid is submitted at  $5\text{€}/\text{MW}$  higher than the forecast<sup>31</sup>. The production allocations are then evaluated with the realized production and prices using the revenue function from equation (63).

---

### Algorithm 2 Deterministic

---

```

for each day  $t^D \in T^D$  do
    Receive forecasts  $\{\hat{\lambda}^F, \hat{\mathbf{p}}^{FC}\}$ 
    Solve model 70-77 to get  $\mathbf{p}^F$  and  $\mathbf{p}^H$ 
    for each hour  $t \in [t^D..t^D+23]$  do
        if  $p_t^F > 0$  then
            Submit production bid of  $(\hat{\lambda}_t^F - 5\text{€}/\text{MW}, p_t^F)$ 
        else
            Submit consumption bid of  $(\hat{\lambda}_t^F + 5\text{€}/\text{MW}, p_t^F)$ 
        end if
    end for
    Realize uncertainties and receive revenue from  $\mathbf{R}(\mathbf{p}^F, \mathbf{p}^H)$ 
end for

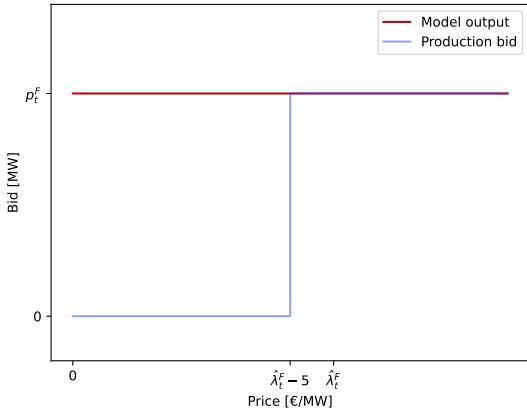
```

---

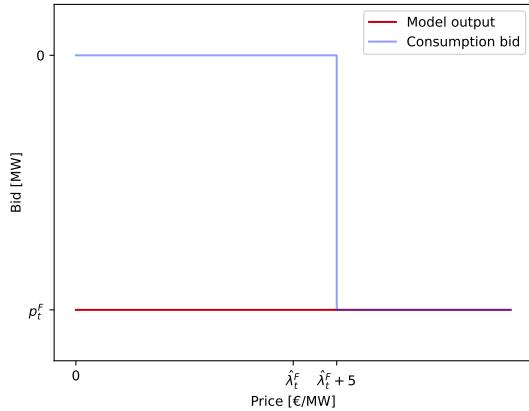
The bidding curve thus becomes a simple step function at the submitted price and the function value  $p_t^F$ , as illustrated in figure 16 and 17 for production and consumption respectively.

---

<sup>31</sup>This value serves as a buffer, and represents roughly one standard deviation of the forecast error, which has a mean of  $-0.78 \approx 0$ .



**Figure 16:** Production bid in forward market

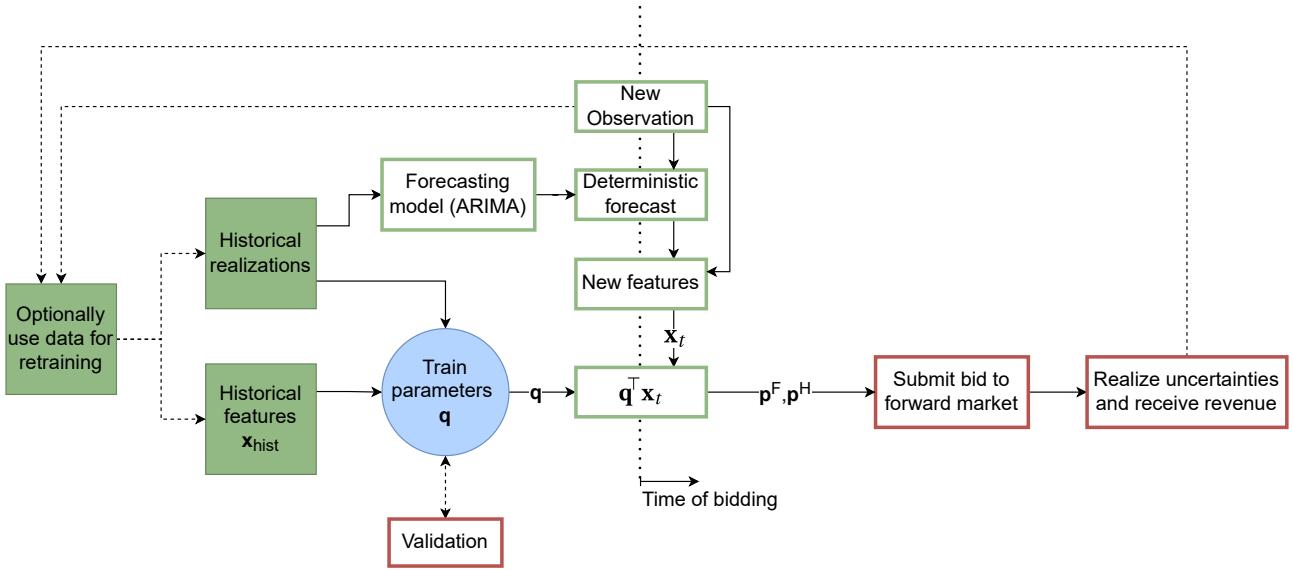


**Figure 17:** Consumption bid in forward market

An alternative and more complex, but perhaps better performing procedure could be to run the algorithm on a series of prices distributed around the forecasted price to generate a price curve instead of a single bid, but this is not a recognized method of operation and is therefore not implemented in the deterministic model. This would begin to resemble a stochastic approach since it technically constitutes a (very simple) modelling of the uncertainty, even though it would be a measure implemented in the algorithm outside the optimization problem. For this reason the deterministic is left as developed, and thereby kept strictly deterministic in its approach.

### 5.3 Feature-Driven Model

The feature-driven model is shown in overview in figure 18. The added detail compared to figure 15 is all the dashed lines, which represent optional additions to the model. Validation of the trained parameters can be anything from a simple feasibility check using a selected set of features, to an out-of-sample testing phase where the model is tested on different data than it was trained on, to ensure that the model generalizes well to new data. Validation is not performed explicitly in this project, although the testing of the models performed in the project could be thought of as validation if a model was applied on new data afterwards. Whenever a model has been used on a new set of features and the uncertain values have been realized, this data constitutes a new datapoint which can be used for retraining the model whenever the producer sees fit.



**Figure 18:** Schematic of the feature-driven model

This section develops the feature-driven model, starting with the optimization problem used for training the parameters, and then delving progressively further into detail about how a forward bid and a hydrogen production schedule is determined from the parameters, and how the parameters are structured.

### 5.3.1 Training Procedure

When developing data-driven models, an important conceptual dichotomy is between batch learning, and online learning. Batch learning considers a dataset as a whole, and optimizes the model based on the entire set of data simultaneously. Online learning updates its parameters for each datapoint sequentially, one at a time<sup>32</sup>. This project uses batch learning as the chosen training approach, and an online learning formulation is left for future work. The dataset considered is split into a training set, and a test set<sup>33</sup>. The training set is used for tuning the model parameters, and the test set is used to emulate real-world application of the model<sup>34</sup>.

The training procedure consists of optimizing an objective function on a dataset, with the model parameters as decision variables. If the data used for training, the model architecture, and the chosen features are appropriate, the resulting values of the model parameters will express relations between the features and the predicted quantities that generalize to new datapoints. The optimization model used for training the parameters is simply the base model (49)-(62), with the forward bid and hydrogen production redefined as the linear combination of the parameters and a feature vector:

<sup>32</sup>Hybrid approaches are also possible, and are especially popular for training neural networks, where the model is trained sequentially using algorithms typical for online learning (often gradient descent methods) on small batches of data. The motivation for using such an approach is the computational efficiency associated with finding the optimal tradeoff between using parallel operations (using batches of data), and simplicity of calculations (gradient descent). In order to generalize the model to the data, the same dataset is often trained on several times (each full run-through called an epoch), and regularization will often be a necessary implementation. When using an optimization problem to perform batch learning, such measures are not applied.

<sup>33</sup>As was described in section 3.3.

<sup>34</sup>Which is how an out-of-sample validation test would be performed.

---


$$p_t = \mathbf{q}^\top \mathbf{x}_t \quad \forall t \in T \quad (78)$$

$$\downarrow$$

$$p_t^F = \mathbf{q}^{F^\top} \mathbf{x}_t \quad \forall t \in T \quad (79)$$

$$p_t^H = \mathbf{q}^{H^\top} \mathbf{x}_t \quad \forall t \in T \quad (80)$$

The training problem for the feature-driven model is then given as:

$$\underset{\mathbf{q}^F, \mathbf{q}^H}{\text{maximize}} \sum_{t \in T^{hist}} \left[ \lambda_t^F \mathbf{q}^{F^\top} \mathbf{x}_t + \lambda_t^H \mathbf{q}^{H^\top} \mathbf{x}_t + \lambda_t^{DW} E_t^{DW} - \lambda_t^{UP} E_t^{UP} \right] \quad (81)$$

s.t.

$$\mathbf{q}^{F^\top} \mathbf{x}_t \leq \bar{P}^F \quad \forall t \in T^{hist} \quad (82)$$

$$\mathbf{q}^{H^\top} \mathbf{x}_t \leq \bar{P}^H \quad \forall t \in T^{hist} \quad (83)$$

$$\mathbf{q}^{F^\top} \mathbf{x}_t \geq -\bar{P}^H \quad \forall t \in T^{hist} \quad (84)$$

$$\mathbf{q}^{H^\top} \mathbf{x}_t \geq 0 \quad \forall t \in T^{hist} \quad (85)$$

$$\sum_{t=t^D}^{t^D+23} \mathbf{q}^{H^\top} \mathbf{x}_t \geq P^{DMin} \quad \forall t^D \in T^{hist,D} \quad (86)$$

$$E_t^{settled} = E_t^{real} - \mathbf{q}^{F^\top} \mathbf{x}_t - \mathbf{q}^{H^\top} \mathbf{x}_t \quad \forall t \in T^{hist} \quad (87)$$

$$E_t^{DW} \geq E_t^{settled} \quad \forall t \in T^{hist} \quad (88)$$

$$E_t^{DW} \leq E_t^{settled} + M \cdot b_t \quad \forall t \in T^{hist} \quad (89)$$

$$E_t^{DW} \leq M \cdot (1 - b_t) \quad \forall t \in T^{hist} \quad (90)$$

$$E_t^{UP} \geq -E_t^{settled} \quad \forall t \in T^{hist} \quad (91)$$

$$E_t^{UP} \leq -E_t^{settled} + M \cdot (1 - b_t) \quad \forall t \in T^{hist} \quad (92)$$

$$E_t^{UP} \leq M \cdot b_t \quad \forall t \in T^{hist} \quad (93)$$

$$E_t^{UP}, E_t^{DW} \in \mathbb{R}^+ \quad \forall t \in T^{hist} \quad (94)$$

$$E_t^{settled} \in \mathbb{R} \quad \forall t \in T^{hist} \quad (95)$$

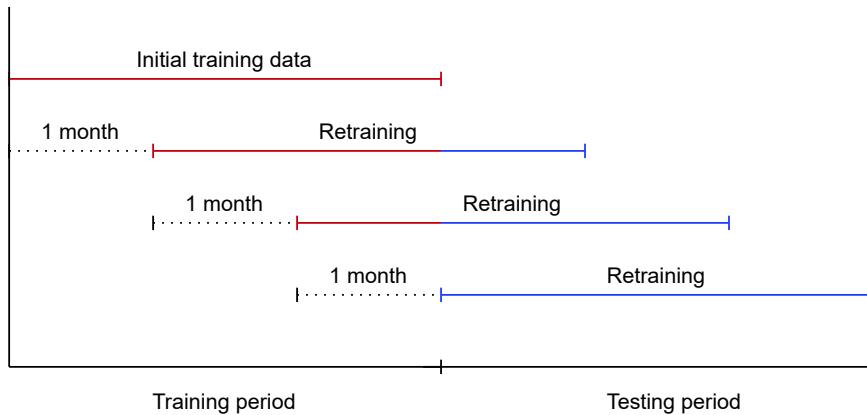
$$\mathbf{q}^F, \mathbf{q}^H \in \mathbb{R} \quad (96)$$

Where  $T^{hist}$  indicates the historical data used for training.

The training data should always be the most recent available data, in order for it to resemble the data it is applied on as closely as possible. The length of the training period  $|T^{hist}|$  then becomes a hyper-parameter, that should be chosen based on out-of-sample test data.

This optimization problem constitutes a complete training of the model. But as soon as the model have been applied on new datapoints and the uncertainties of these have been revealed, they classify as historical samples and could be utilized in training the model parameters. This calls for a retraining of the model, with the new datapoints included in the original dataset. Given  $|T^{hist}|$  as a fixed hyper-parameter, including the new datapoint in the training set should entail removing the oldest datapoint, such as to keep  $|T^{hist}|$  constant. The data used for training is thus a sliding window of data, as illustrated using a monthly retraining procedure in figure 19. The training period should be kept constant because the length of the training

period found by out-of-sample investigation is reflecting the rate of diminishing dependency on older patterns in the data. Any performance improvement resulting from varying the length of the training period would represent one of the following cases: 1) An inaccurate result from the out-of-sample test of what the optimal training period actually is, 2) A statistical coincidence, or 3) The dependency on older data patterns (the stationarity) actually changes throughout the period being investigated. It will be very hard to distinguish which case(s) is true, but the conclusions to be drawn are very different. Varying the training period by not keeping  $|T^{hist}|$  constant is therefore not recommended, since the results could very easily lead to wrong conclusions being drawn. The length of the sliding window therefore remains  $|T^{hist}|$  throughout the retraining process.



**Figure 19:** Schematic of sliding window approach

The frequency of retraining then becomes yet another hyper-parameter. This project employs a retraining of the model each month, to keep computation times below 8 hours on the used machine<sup>35</sup>.

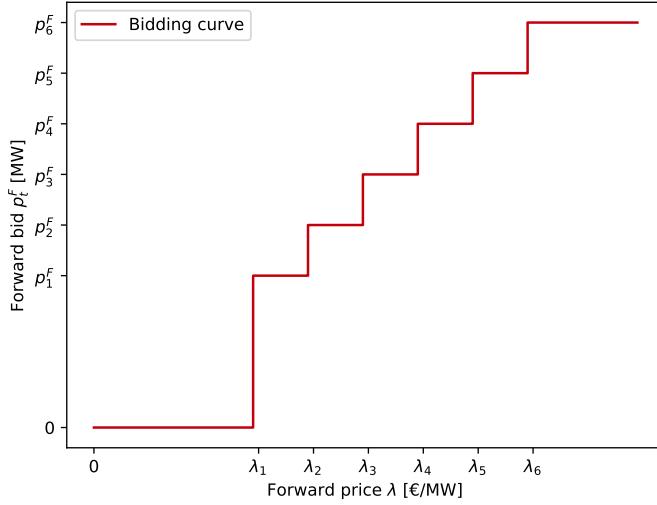
### 5.3.2 Constructing the Bidding Curve

The choice of features represents the awareness of the model, so if the model should account for some specific type of data, it must be included in the feature vector  $\mathbf{x}_t$ .

The most obvious feature to include, is the production forecast. The clearing price in the forward market would be very valuable information as well, since the producer would prefer to allocate most if not all production to the electrolyzer when the forward price is low, and allocate most production to a forward market bid when the price is high. This could be achieved to some degree by utilizing a price forecast as done in the deterministic approach. Another possibility however, is utilizing the fact that the forward market bids are submitted as a punctuated curve, not as a single value. An example of such a curve is shown in figure 20, where the curve consists of a series of six bids  $\{(\lambda_i, p_i^F) | i \in [1..6]\}$ . This means that the model can submit bids with different amounts of production allocated to the forward market bid, conditional on what the clearing price will be. Note that this is possible because the scheduled allocation to the electrolyzer is *not* committed, and can be changed at any time up to the time of hydrogen production<sup>36</sup>, meaning it is free to be decided *after* the forward market is cleared.

<sup>35</sup>See section A.1 in Appendix for the machine specifications

<sup>36</sup>Extending even to real-time adjustment as described in section 4.4.



**Figure 20:** Example of a bidding curve in the forward market

A naive approach to modelling such a price curve would be to assign a parameter value  $q_t^\lambda$  to all the (pre-decided) price points  $\lambda_i$ , and have it output the value of  $p_i^F$ . However, this would quickly result in a lot of modelling parameters. Adding parameters to a model, and thereby increasing the complexity, does not only increase the computational effort required to train the model, it also increases the amount of training data required to train the model and introduces the risk of overfitting. To utilize the available training data as efficiently as possible, it is therefore preferable to have as few adjustable parameters as possible. Overfitting occurs when the model contains more parameters to be adjusted, than the actual data relation being investigated can justify. This means that the model adjusts to random fluctuations in the data as if it were actual patterns, resulting in poor performance on new data that hasn't been used for training the model. Because the models are tested on a different dataset than what was used for training, any potential overfitting will be revealed during the evaluation.

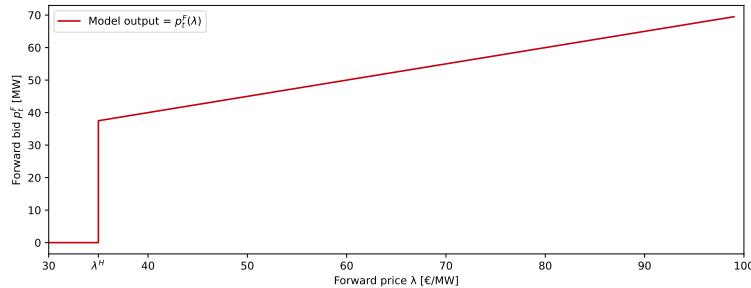
The present subsection will develop a much more parsimonious approach to the dependency on the forward price, with an equivalent bidding flexibility.

Let us first define the allocated production as a function of the (not-yet-realized) forward price, in the simplest case a linear function:

$$p_t^F(\lambda_t^F) = a_t^F \lambda_t^F + c_t^F \quad (97)$$

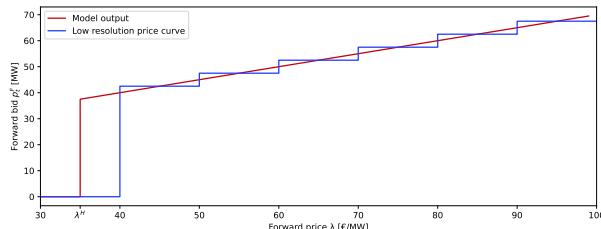
$$p_t^H(\lambda_t^F) = a_t^H \lambda_t^F + c_t^H \quad (98)$$

With the function defined on some interval,  $[\lambda_{min}, \lambda_{max}]$ . This is illustrated in figure 21, where a the linear function is defined on the interval  $[\lambda^H, \infty)$ , and the function takes the value zero outside this interval.

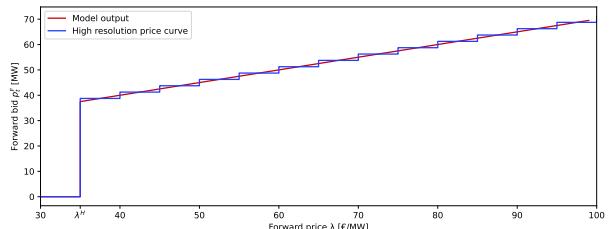


**Figure 21:** Linear function as model output

From this curve, a punctuated bidding curve can be constructed based on the allowed resolution of the bidding curve. The meaning of resolution being how small price intervals are allowed, where figure 22 shows a low resolution of 1 bid pr.  $10 \frac{\text{€}}{\text{MW}}$ , and figure 23 shows a higher resolution of 1 bid pr.  $5 \frac{\text{€}}{\text{MW}}$ . This is meant to show that the described approach can be accommodated to whatever resolution the market operator allows, and an appropriate bidding curve can be constructed.



**Figure 22:** Low resolution bidding curve



**Figure 23:** High resolution bidding curve

The functions  $p_t^F(\lambda_t^F)$  and  $p_t^H(\lambda_t^F)$  from equations (97) and (98) should now be elaborated, and since both functions have the same structure, the superscript is dropped for notational convenience in the following development. Returning to the simple formulation introduced in equations (79) and (80), the production allocation can be rewritten as the sum of scalar multiplications:

$$p_t = \mathbf{q}^\top \mathbf{x}_t = \sum_{n=1}^{|\mathbf{q}|} q^n x_t^n \quad (99)$$

Ideally, the forward price would be included in the feature vector  $\mathbf{x}_t$ , along with the other relevant features. Since the forward price is not known at the time of computation this seems to be impossible, but a work-around can be found using the described approach of constructing a price curve. Considering a simple example where only a production forecast and the forward price are included as features  $x_t^{prod}$  and  $x_t^\lambda$  respectively, the scalar multiplication from equation (99) becomes:

$$\mathbf{q}^\top \mathbf{x}_t = q^{prod} x_t^{prod} + q^\lambda x_t^\lambda \quad (100)$$

The first term is readily computed since both  $\mathbf{q}$  and  $x_t^{prod}$  is known. By simple change of notation it is seen how leaving  $x_t^\lambda$  unknown, this corresponds to  $p_t(\lambda)$ :

---


$$x_t^\lambda = \lambda_t^F \quad (101)$$

$$a_t = q^\lambda \quad (102)$$

$$c_t = q^{prod} x_t^{prod} \quad (103)$$

↓

$$\mathbf{q}^\top \mathbf{x}_t = q^{prod} x_t^{prod} + q^\lambda x_t^\lambda \quad (104)$$

$$\mathbf{q}^\top \mathbf{x}_t = c_t + a_t \lambda_t^F \quad (105)$$

Which is the exact expression used in equations (97) and (98).

This method generalizes to any number of features. As long as the remaining features are known at the time of computation, the result can be gathered under the constant  $c_t$ , and used directly in equations (97) and (98). Furthermore,  $a_t$  is not restricted to consist of a single  $q^\lambda$  parameter, but can instead be a vector of values  $\mathbf{q}^\lambda$ . This is shown in the following relations, where  $\mathbf{1}$  denotes a vector of 1's, resulting  $\mathbf{q}^{\lambda\top} \mathbf{1}$  to be the summation of the parameters of  $\mathbf{q}^\lambda$ .

$$\mathbf{q}^{\lambda\top} \mathbf{1} x_t^\lambda = q^{\lambda,1} x_t^\lambda + q^{\lambda,2} x_t^\lambda + q^{\lambda,3} x_t^\lambda \quad (106)$$

$$\mathbf{q}^{\lambda\top} \mathbf{1} x_t^\lambda = (q^{\lambda,1} + q^{\lambda,2} + q^{\lambda,3}) x_t^\lambda \quad (107)$$

$$a_t = (q^{\lambda,1} + q^{\lambda,2} + q^{\lambda,3}) \quad (108)$$

↓

$$\mathbf{q}^{\lambda\top} \mathbf{1} x_t^\lambda = a_t x_t^\lambda \quad (109)$$

To utilize this approach,  $\lambda$  will therefore be included as a an unknown variable in the feature vector  $\mathbf{x}_t$ , such that the model outputs a linear function of the price, which will then be discretized to construct a bidding curve. The actual forward bid is then determined by the realized price in the forward market, where the accepted bid is the function value of the price curve at the realized forward price.  $\lambda$  appearing as an unknown variable in the feature vector will be denoted as the "price variable", and will be treated the same as other features when multiplying with the parameters  $\mathbf{q}$ .

When training the model using (81)-(96), the only relevant value on the constructed price curve is the bid that was actually accepted, which depends on the realized forward price, since the remaining parts of the price curve is not realized and therefore not included in the objective in equation (81). When performing the training on historical data, the historical forward price can therefore be used directly, since this corresponds to the realization of the price curve that the parameters would adjust to<sup>37</sup>. The dependency on  $\lambda$  and the price curve construction is therefore not included in the formulation of the training problem (81)-(96).

### 5.3.3 The Feature-Driven Algorithm

The core elements of the feature-driven approach has now been developed, and a generic algorithm can be defined for the feature-driven models, to show the steps involved in every model implementation. The feature-driven model itself consists of the parameters  $\mathbf{q}^F$  and  $\mathbf{q}^H$ , along with the structure and content of the feature vector  $\mathbf{x}_t$ . The generic feature-driven algorithm therefore remains agnostic to the architecture of the parameters and the features. This means that when exploring different model architectures, training

<sup>37</sup>The prices are rounded to two decimal places, and the allowed bidding resolution is expected to be the same, and therefore the prices can be used directly. If the allowed decimal resolution on the forward bids is not greater than or equal to the precision of the price realization, the curve would need to be constructed.

---

periods, feature-spaces and other high-level decisions and hyper-parameters, the procedure will always follow the basic steps described in algorithm 4. The dataset used for training consists of historical feature vectors and their associated realizations, and is denoted  $\mathcal{T}$ . After the model has been trained, it is applied to the feature vector at each new time step to decide the forward bid and hydrogen production. The resulting model output is bounded by the feasibility sets  $F$  and  $H$  for forward market bids and hydrogen production schedules respectively, since feasibility is not guaranteed by the model for new feature vectors outside the training environment.

$$F = \left\{ p_t^F : -\bar{P}^H \leq p_t^F \leq \bar{P}^F \right\} \quad (110)$$

$$H = \left\{ p_t^H : 0 \leq p_t^H \leq \bar{P}^H \right\} \quad (111)$$

The functions  $\alpha^F[\cdot]$  and  $\alpha^H[\cdot]$  ensures this feasibility for forward bids and hydrogen production respectively:

$$\alpha^F \left[ \mathbf{q}^{F^\top} \mathbf{x}_t \right] = \begin{cases} -\bar{P}^H & \text{if } \mathbf{q}^{F^\top} \mathbf{x}_t < -\bar{P}^H \\ \bar{P}^F & \text{if } \mathbf{q}^{F^\top} \mathbf{x}_t > \bar{P}^F \\ \mathbf{q}^{F^\top} \mathbf{x}_t & \text{else} \end{cases} \quad (112)$$

$$\alpha^H \left[ \mathbf{q}^{H^\top} \mathbf{x}_t \right] = \begin{cases} 0 & \text{if } \mathbf{q}^{H^\top} \mathbf{x}_t < 0 \\ \bar{P}^H & \text{if } \mathbf{q}^{H^\top} \mathbf{x}_t > \bar{P}^H \\ \mathbf{q}^{H^\top} \mathbf{x}_t & \text{else} \end{cases} \quad (113)$$

The bidding curve is constructed from the feasible outputs  $p_t^F(\lambda) = \alpha^F \left[ \mathbf{q}^{F^\top} \mathbf{x}_t \right]$ , by evaluating  $p_t^F(\lambda)$  for each value of  $\lambda$  in the allowed resolution of price values, as described in the previous subsection 5.3.2. The price cleared in the forward market determines the bid that gets accepted, and consequently also the scheduled hydrogen production, where the realized final values are denoted by  $p_t^F(\lambda_t^F)$  and  $p_t^H(\lambda_t^F)$ .

---

### Algorithm 3 Feature-Driven

---

**Require:** Historical data  $\mathcal{T}$

Solve (81)-(96) on data  $\mathcal{T}$  to get  $\mathbf{q}^F$  and  $\mathbf{q}^H$

**for** each hour  $t \in T$  **do**

    Receive feature vector  $\mathbf{x}_t$

$$p_t^F(\lambda) = \alpha^F \left[ \mathbf{q}^{F^\top} \mathbf{x}_t \right]$$

$$p_t^H(\lambda) = \alpha^H \left[ \mathbf{q}^{H^\top} \mathbf{x}_t \right]$$

    Construct bidding curve from  $p_t^F(\lambda)$  and submit

    Cleared forward market determines  $p_t^F(\lambda_t^F)$  and  $p_t^H(\lambda_t^F)$

    Realize uncertainties and receive revenue from  $\mathbf{R}(p_t^F(\lambda_t^F), p_t^H(\lambda_t^F))$

**end for**

---

---

**Algorithm 4** Feature-Driven

---

**Require:** Historical data  $\mathcal{T}$ Solve training model on historical data  $\mathcal{T}$  to get  $\mathbf{q}^F$  and  $\mathbf{q}^H$ **for** each hour  $t \in T$  **do**    Receive feature vector  $\mathbf{x}_t$ 

$$p_t^F(\lambda) = \alpha^F [\mathbf{q}^{F\top} \mathbf{x}_t]$$

$$p_t^H(\lambda) = \alpha^H [\mathbf{q}^{H\top} \mathbf{x}_t]$$

    Construct bidding curve from  $p_t^F(\lambda)$  and submit    Cleared forward market determines  $p_t^F(\lambda_t^F)$  and  $p_t^H(\lambda_t^F)$     Realize uncertainties and receive revenue from  $\mathbf{R}(p_t^F(\lambda_t^F), p_t^H(\lambda_t^F))$ **end for**

---

Algorithm 6 adds a retraining procedure, where the model is retrained each new month. The sliding window of training data<sup>38</sup> is achieved by updating the dataset used for training  $\mathcal{T}$  at the end of each month.  $T^m$  denotes that the time steps are divided into monthly intervals  $m$ .

---

**Algorithm 5** Feature-Driven retraining

---

**Require:** Historical data  $\mathcal{T}$ **for** each month  $m \in T^m$  **do**    Solve (81)-(96) on data  $\mathcal{T}$  to get  $\mathbf{q}^F$  and  $\mathbf{q}^H$     **for** each hour  $t \in m$  **do**        Receive feature vector  $\mathbf{x}_t$ 

$$p_t^F(\lambda) = \alpha^F [\mathbf{q}^{F\top} \mathbf{x}_t]$$

$$p_t^H(\lambda) = \alpha^H [\mathbf{q}^{H\top} \mathbf{x}_t]$$

        Construct bidding curve from  $p_t^F(\lambda)$  and submit        Cleared forward market determines  $p_t^F(\lambda_t^F)$  and  $p_t^H(\lambda_t^F)$         Realize uncertainties and receive revenue from  $\mathbf{R}(p_t^F(\lambda_t^F), p_t^H(\lambda_t^F))$     **end for**    Append  $|m|$  new datapoints to  $\mathcal{T}$     Remove  $|m|$  oldest datapoints from  $\mathcal{T}$ **end for**

---

<sup>38</sup>As was illustrated in 19.

---

**Algorithm 6** Feature-Driven retraining

---

**Require:** Historical data  $\mathcal{T}$

**for** each month  $m \in T^m$  **do**

Solve training model on historical data  $\mathcal{T}$  to get  $\mathbf{q}^F$  and  $\mathbf{q}^H$

**for** each hour  $t \in m$  **do**

Receive feature vector  $\mathbf{x}_t$

$p_t^F(\lambda) = \alpha^F [\mathbf{q}^{F\top} \mathbf{x}_t]$

$p_t^H(\lambda) = \alpha^H [\mathbf{q}^{H\top} \mathbf{x}_t]$

Construct bidding curve from  $p_t^F(\lambda)$  and submit

Cleared forward market determines  $p_t^F(\lambda_t^F)$  and  $p_t^H(\lambda_t^F)$

Realize uncertainties and receive revenue from  $\mathbf{R}(p_t^F(\lambda_t^F), p_t^H(\lambda_t^F))$

**end for**

Append  $|m|$  new datapoints to  $\mathcal{T}$

Remove  $|m|$  oldest datapoints from  $\mathcal{T}$

**end for**

---

### 5.3.4 Domain Specific Parameters

The concept of domain specific parameters is to have different sets of parameters covering different domains of inputs. One type of domain specific parameters could be a "high production forecast" model, and a "low production forecast" model, where the model used depends on the value of the forecast received. Another possibility is using different price domains, which will be used in this project, and developed in the present section. Since equations (97) and (98) are constrained to be defined on a specific price interval, the researcher is not limited to using only a single function for the entire bidding curve, but can instead develop different models for different price intervals as well. A natural threshold where it might be beneficial to divide the model into different price domains is at the hydrogen price. Such a threshold can be achieved by defining two bidding functions  $f^1$  and  $f^2$  where:

$$\text{dom}(f^1) = [0, \lambda^H] \quad (114)$$

$$\text{dom}(f^2) = (\lambda^H, +\infty) \quad (115)$$

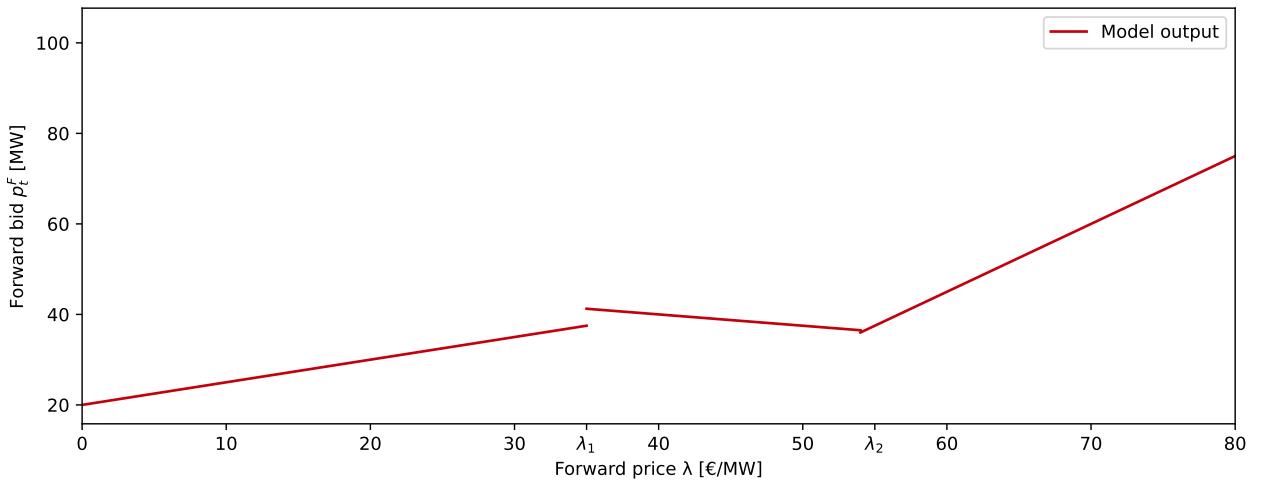
Note that  $[\cdot, \cdot]$  denotes a closed interval of real numbers,  $(\cdot, \cdot)$  denotes an open interval of real numbers, and  $\text{dom}(f)$  denotes the domain of a function  $f$ . Note also that splitting the functions into domains as described here results in the amount of parameters being multiplied by the amount of domains. However, because the domains<sup>39</sup> can be decided based on statistical characteristics of the data, a significantly better trade-off between complexity and flexibility can be achieved than when the amount of parameters was dependent on the resolution of the price bids, as was introduced as the naive approach in the beginning of subsection 5.3.2.

The bidding curve resulting from a set of bidding functions will be a discontinuous piecewise linear function, as illustrated in figure 24. It should be noted that this could potentially result in unstable models, where a small change in price around the domain boundaries of the functions would result in a large change in bid size<sup>40</sup>. Fortunately this is fairly easy to test, and would be a sign that the domains are chosen inappropriately, since the actual optimal model output should be approximately the same for a small change in price, which the model will reflect if designed correctly.

---

<sup>39</sup>And thereby the degree of complexity.

<sup>40</sup>An example of this is seen at the price  $\lambda_1$ , where the bid varies by several MW on either side of the price point.



**Figure 24:** Price curve resulting from three price domains

Three price domains are used in total:

$$\text{domain} = \begin{cases} 1, & \lambda < \lambda^H \\ 2, & \lambda^H < \lambda < \lambda^{90\%} \\ 3, & \text{else} \end{cases} \quad (116)$$

Where  $\lambda^{90\%} = 53.32\text{€}/\text{MWh}$  is the 90 % quantile of the forward price for the entire training set. This value is chosen such that the parameters that will be used for the majority of the data (domains 1 and 2) will not be fitted to the 10% highest prices that most likely have a high degree of outliers. The concrete implementation in the parameters  $\mathbf{q}$  will be presented for each of the feature-driven models in section 5.5.

### 5.3.5 Model Architecture

The architecture of a model refers to the structure of the parameters  $\mathbf{q}$ , and how they relate to the feature vector  $\mathbf{x}_t$ . Two different architectures are tested in this project, a *General* architecture with a single  $q$ -parameter for each feature, and an *Hourly* architecture with a different  $q$ -parameter depending on hour of bidding.

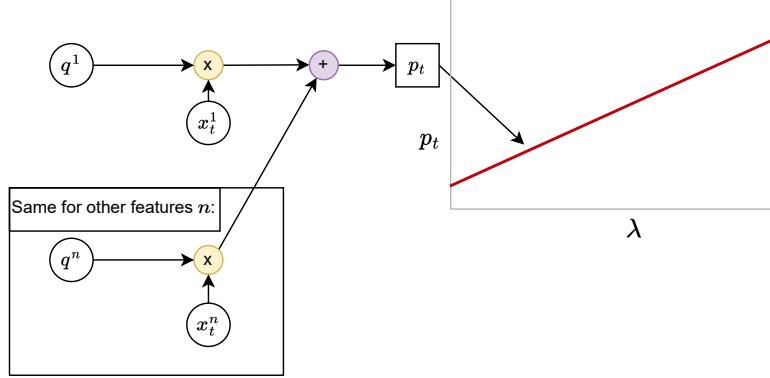
There is no difference between the how  $p_t^F$  and  $p_t^H$  are attained for either of the architectures, and so the superscript is dropped in the development of the architectures. The specific feature vector is not relevant in this section, as the developed architecture will be applied identically to all features<sup>41</sup>. The price variable  $\lambda$  that is included in the feature vector as an unknown to construct the price curve will not be shown explicitly, but will be treated the same as the remaining features.

The *General* architecture consists of a single parameter for each feature. The model output is given by each parameter multiplied with the associated feature, summed over all  $N$  features:

$$p_t^{\text{General}} = \sum_{n=1}^N q^n x_t^n \quad (117)$$

<sup>41</sup>Each feature will either have one general  $q$ -parameter, or an hourly  $q$ -parameter.

Figure 25 shows a schematic of the *General* architecture. The yellow (x)-combiners indicate multiplication, and the purple (+)-combiners indicate addition. The schematic explicitly shows the multiplication for the first feature, which is added with the parameter multiplication for the other features to create the price curve.



**Figure 25:** Schematic overview of *General* architecture

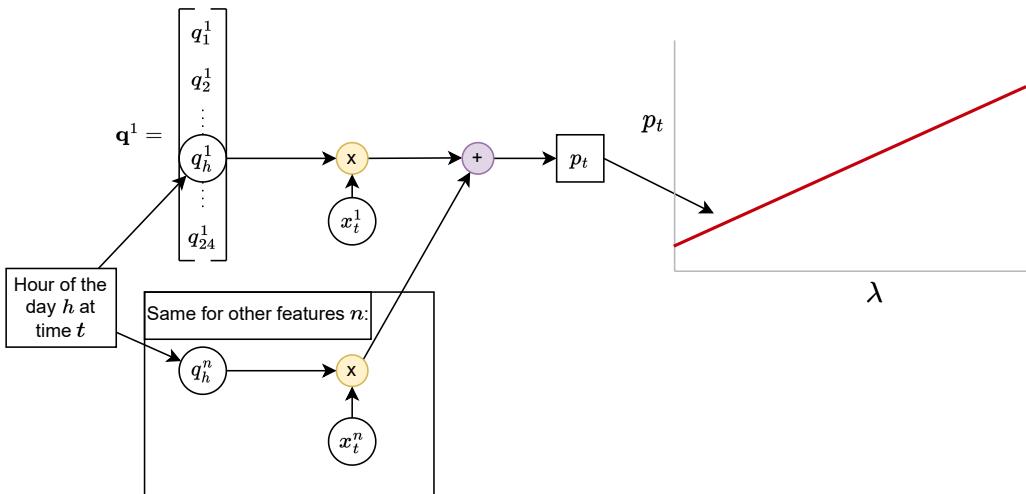
The *Hourly* architecture has a vector of 24 parameters for each feature, each parameter associated with a specific hour of the day. The hour of the day is calculated at each hour  $t$  by using the modulo operator:

$$h_t = \text{mod}(t, 24) \quad (118)$$

The variable  $h_t$  will then take a value in the range [0..23], which will be used for indexing the appropriate parameter for the given time step. At each time step, the hour of the day then determines which element of the parameter vector should be multiplied with the feature:

$$p_t^{Hourly} = \sum_{n=1}^N q_h^n x_t^n \quad (119)$$

This allows the model to adjust to the patterns in the data conditional on the time of day. Figure 26 shows a schematic of the *Hourly* architecture. The schematic explicitly shows the selection of the parameter specific to the hour of the day, which is used in the parameter multiplication. As in the *General* architecture, this result is added with the result from all the remaining features to construct the price curve.



**Figure 26:** Schematic overview of *Hourly* architecture

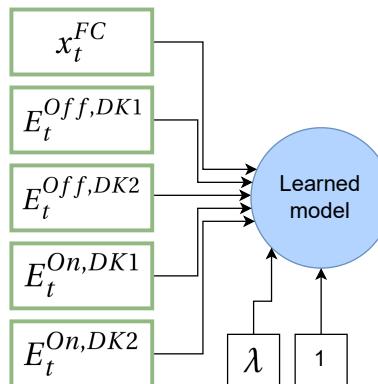
## 5.4 The Feature Vector

The feature vector determines what information the feature-driven model is exposed to. The available features each timestep are all related to the realized production, and includes the production forecast  $P^W(v_t)$  from SG, and the aggregated wind production forecasts from Energinet  $E_t^{Off,DK1}$ ,  $E_t^{Off,DK2}$ ,  $E_t^{On,DK1}$ , and  $E_t^{On,DK2}$ .  $P^W(v_t)$  will be denoted as  $x_t^{FC}$  from hereon, to indicate its presence as a feature. A dependency on the forward price is given by including an unknown price variable in the feature vector as described in section 5.3.2. Three different feature vectors will be introduced: (1) One with all the available features included, (2) one with all the production related features combined to a single feature, and (3) one without the aggregated forecasts. This will allow an assessment of the value of the different features and approaches as they are tested through the models.

The first feature vector to be introduced contains all the features. In addition to the production features, a bias<sup>42</sup> is included by adding the constant value 1 as a feature. Finally,  $\lambda$  is included as the price variable, resulting in a feature vector using All Features<sup>43</sup>.

$$\mathbf{x}_t^{AF} = \begin{bmatrix} x_t^{FC} \\ E_t^{Off,DK1} \\ E_t^{Off,DK2} \\ E_t^{On,DK1} \\ E_t^{On,DK2} \\ 1 \\ \lambda \end{bmatrix} \quad (120)$$

The model is thus exposed to all the features, as illustrated in figure 27.



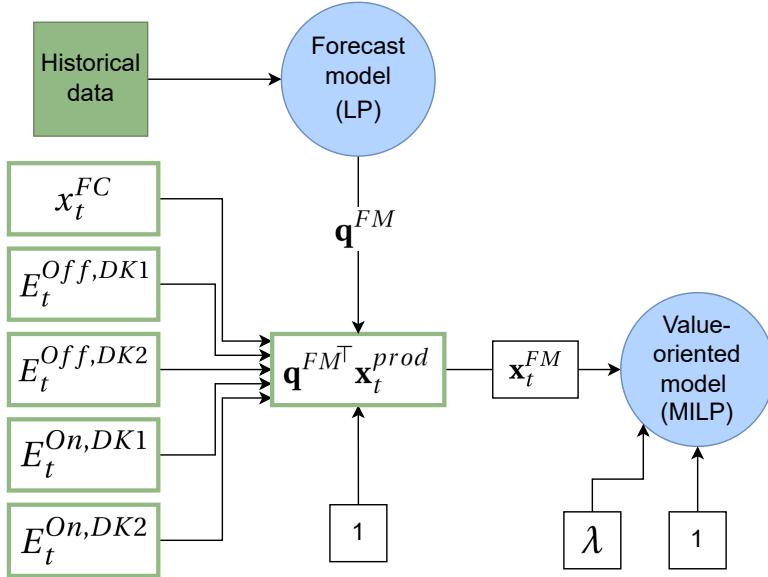
**Figure 27:** Schematic of all features being exposed to a model

The second feature vector to be introduced combines the features to produce a single new feature. Inspired by the approach in [12], a separate feature-driven model can be created to collect the production related features into a single improved forecast, instead of having each feature separate in the final feature vector. The approach is illustrated in figure 28. The reasoning behind creating such a forecasting model is firstly that the relations between production related features and realized production is not expected to be subject to short-term patterns and variations as can be the case with price data, and can therefore be trained using older data, meaning a larger set of datapoints can be used. It is thereby a way to decouple the

<sup>42</sup>A constant value of displacement in the model.

<sup>43</sup>The first letters are capitalized and in bold font to indicate the source of the abbreviation used in the superscript of the associated feature vector, as will be done with the remaining feature vectors as well.

difference in historical dependency between price and production, thus allowing for the long-term dependency in the production data to be captured, while simultaneously using short training periods to capture the non-stationarity of the price data. An added benefit is that when training, the forecasting model is an LP, whereas the value-oriented models are MILPs. The computational effort can thus be reduced because several parameters are moved from a MILP to an LP.



**Figure 28:** Schematic of a forecast model combining the different features

The improved forecast is created by defining a new feature-driven model, that is trained to predict the realized production based on a production specific feature vector  $\mathbf{x}_t^{prod}$  shown in equation (121). The trained parameters are then used to combine the features in  $\mathbf{x}_t^{prod}$  into a single improved forecast  $x_t^{FM}$ , which can be used when training the bidding models.

$$\mathbf{x}_t^{prod} = \begin{bmatrix} x_t^{FC} \\ E_t^{Off,DK1} \\ E_t^{Off,DK2} \\ E_t^{On,DK1} \\ E_t^{On,DK2} \\ 1 \end{bmatrix} \quad (121)$$

$$x_t^{FM} = \mathbf{q}^{FM\top} \mathbf{x}_t^{prod} \quad (122)$$

The feature vector  $\mathbf{x}_t^{FM}$  resulting from using the **Forecasting Model** as an intermediate step thus becomes:

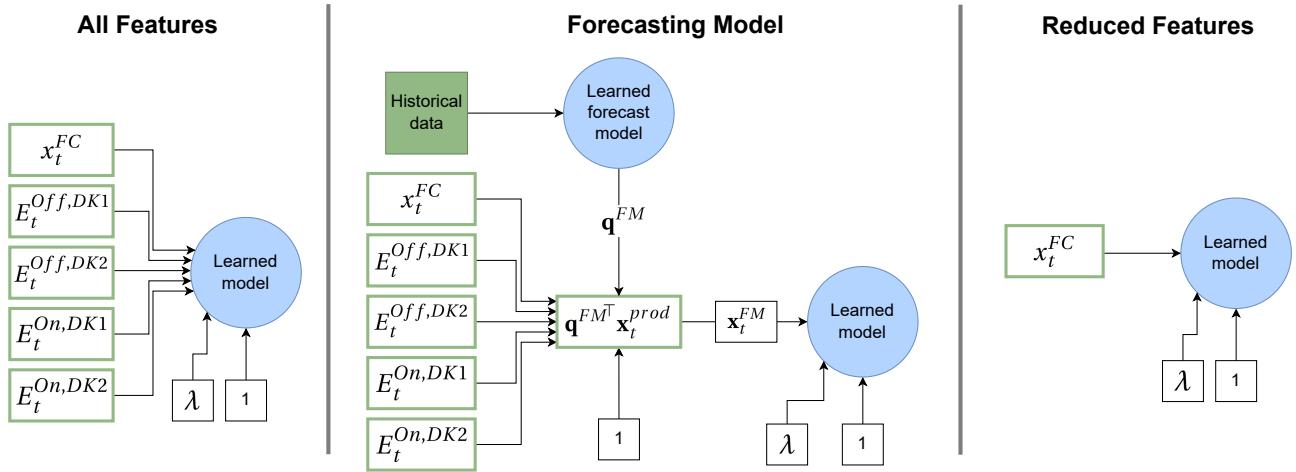
$$\mathbf{x}_t^{FM} = \begin{bmatrix} x_t^{FM} \\ 1 \\ \lambda \end{bmatrix} \quad (123)$$

Subsection 5.4.1 will develop the mathematical formulation of the forecasting model.

To evaluate the benefit of adding the information contained in the aggregated forecasts, a final feature vector is defined without the aggregated forecasts, denoted the **Reduced Feature vector**:

$$\mathbf{x}_t^{RF} = \begin{bmatrix} x_t^{FC} \\ 1 \\ \lambda \end{bmatrix} \quad (124)$$

A schematic of the RF vector is seen long with the other schematics for reference in figure 29.



**Figure 29:** Overview of feature vectors

Each type of model will be tested with the different feature vectors, in order to evaluate if there is any benefit to be found from including the aggregated forecasts, and if there is any benefit from using a forecasting model as an intermediate step.

#### 5.4.1 Forecasting Model

The forecasting model is trained specifically only to predict the realized production from the set of features. The objective of the model is therefore to minimize the deviation between the produced forecast  $\mathbf{q}^{FM\top} \mathbf{x}_t^{prod}$  and the realized production  $E_t^{real}$ :

$$\underset{\mathbf{q}^{FM}}{\text{minimize}} \quad (E_t^{real} - \mathbf{q}^{FM\top} \mathbf{x}_t^{prod})^+ + (\mathbf{q}^{FM\top} \mathbf{x}_t^{prod} - E_t^{real})^+ \quad (125)$$

The  $(\cdot)^+$  operators will be handled by the variables  $u_t$  and  $o_t$  indicating underproduction and overproduction respectively. Since both variables are minimized in the objective, only a lower bound is needed, which should ensure that the variables are equal to either the associated deviation, or zero:

$$u_t \geq \mathbf{q}^{FM\top} \mathbf{x}_t^{prod} - E_t^{real} \quad \forall t \in T \quad (126)$$

$$u_t \geq 0 \quad \forall t \in T \quad (127)$$

$$o_t \geq E_t^{real} - \mathbf{q}^{FM\top} \mathbf{x}_t^{prod} \quad \forall t \in T \quad (128)$$

$$o_t \geq 0 \quad \forall t \in T \quad (129)$$

The complete forecasting model thus becomes:

---


$$\underset{\mathbf{q}^{FM}}{\text{minimize}} \quad u_t + o_t \quad (130)$$

s.t.

$$u_t \geq \mathbf{q}^{FM\top} \mathbf{x}_t^{prod} - E_t^{real} \quad \forall t \in T \quad (131)$$

$$o_t \geq E_t^{real} - \mathbf{q}^{FM\top} \mathbf{x}_t^{prod} \quad \forall t \in T \quad (132)$$

$$o_t, u_t \in \mathbb{R}^+ \quad \forall t \in T \quad (133)$$

$$\mathbf{q}^{FM} \in \mathbb{R} \quad (134)$$

The model is used in algorithm 7 to produce the improved production forecast  $x_t^{FM}$  from the production features  $\mathbf{x}_t^{prod}$ .

---

#### Algorithm 7 Forecasting model

**Require:** Historical data  $\mathcal{T}$

Solve (130)-(134) on data  $\mathcal{T}$  to get  $\mathbf{q}^{FM}$

**for** each  $t \in T$  **do**

    Receive feature vector  $\mathbf{x}_t^{prod}$

    Output  $x_t^{FM} = \mathbf{q}^{FM\top} \mathbf{x}_t^{prod}$

**end for**

---

The forecasting model will use the entire training period for learning the parameters  $\mathbf{q}^{FM}$ .

## 5.5 Final Models

This section presents all the feature-driven models that will be tested in the project. Each model will be presented with an associated algorithm detailing the implementation.

Each of the two architectures will be tested both with and without domain specific parameters. The domains will be for the forward price, where all the model parameters will be specific to a price domain, with three price domains in total, resulting in a piece-wise linear function output from the models as described in subsection 5.3.4. In addition to the price domains, the models will be tested with and without a monthly retraining, to evaluate the effect of this measure as well. This results in six different feature-driven models being tested:

- **GA** - General architecture
- **HA** - Hourly architecture
- **GAPD** - General architecture, 3 price domains
- **HAPD** - Hourly architecture, 3 price domains
- **GAPDR** - General architecture, 3 price domains, monthly retraining
- **HAPDR** - Hourly architecture, 3 price domains, monthly retraining

Each type of model will be tested with three different feature vectors, with a training period varying from 1-12 months. The results will be denoted by each of the abbreviations combined, such that for example the **GAPD** model using the Forecast Model feature vector and trained for **8** months is denoted **GAPD-FM-8**.

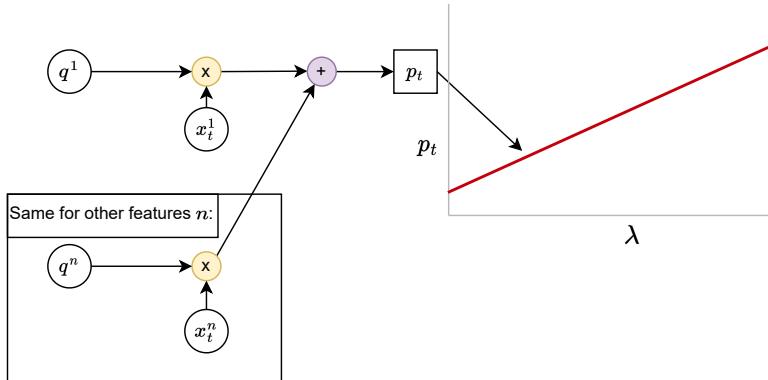
---

Since all three feature vectors **AF**, **FM** and **RF** are applied to each model, a general notation of the feature vector  $\mathbf{x}_t$  is used in the following detailed description of the models, with  $x_t^n$  denoting feature  $n$  of the vector.

Note that the generic algorithms presented in algorithms 4 and 6 served as an overview of the entire process of implementing a learned model starting with training and then application, whereas the algorithms provided in this section serve to explicitly show the implementation of the specific models. The algorithms therefore describe the steps involved for each time step, regardless of whether the time step is within a training or a testing context. The exception being the retraining models, as will be made clear in the associated subsections.

### 5.5.1 GA

The *General* architecture constitutes the most basic of the feature-driven models. The structure of the *General* model is repeated in figure 30 for the readers convenience. Note the superscript differentiating forward bids and hydrogen are dropped in the schematics because the structure is identical for both calculations.



**Figure 30:** Schematic overview of *General* architecture

The operation follows algorithm 8.

---

#### Algorithm 8 GA

```

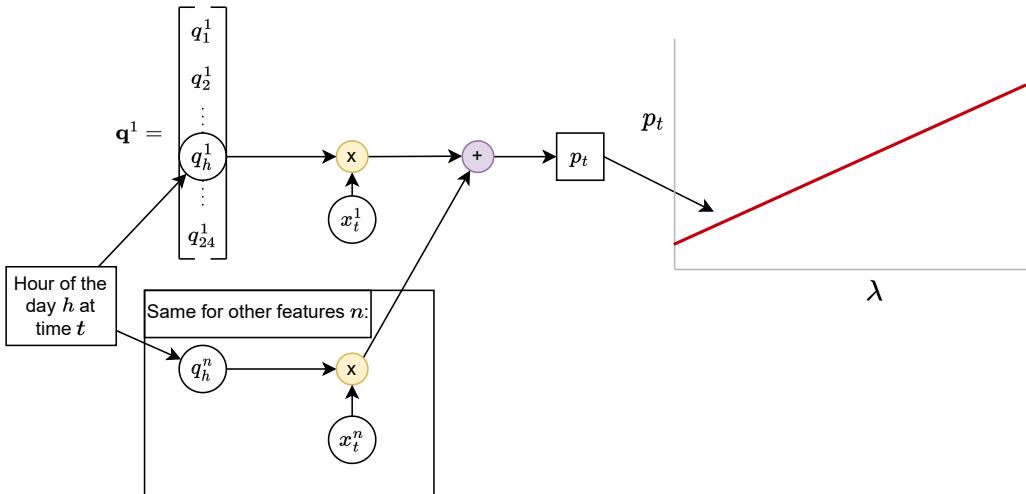
for each hour  $t \in T$  do
    Receive feature vector  $\mathbf{x}_t$ 
     $p_t^F(\lambda) = \alpha^F [\mathbf{q}^{F\top} \mathbf{x}_t]$ 
     $p_t^H(\lambda) = \alpha^H [\mathbf{q}^{H\top} \mathbf{x}_t]$ 
    Construct bidding curve from  $p_t^F(\lambda)$  and submit
    Cleared forward market determines activated forward bid  $p_t^F(\lambda_t^F)$  and hydrogen  $p_t^H(\lambda_t^F)$ 
    Realize uncertainties and receive revenue from  $\mathbf{R}(p_t^F(\lambda_t^F), p_t^H(\lambda_t^F))$ 
end for

```

---

### 5.5.2 HA

The *Hourly* architecture is tested in model **HA**, with the structure repeated in figure 31 for the readers convenience.



**Figure 31:** Schematic overview of *Hourly* architecture

The operation follows algorithm 9, which implements the indexing depending on the time of day, and picks the appropriate set of parameters  $\mathbf{q}$  accordingly.

---

#### Algorithm 9 HA

---

```

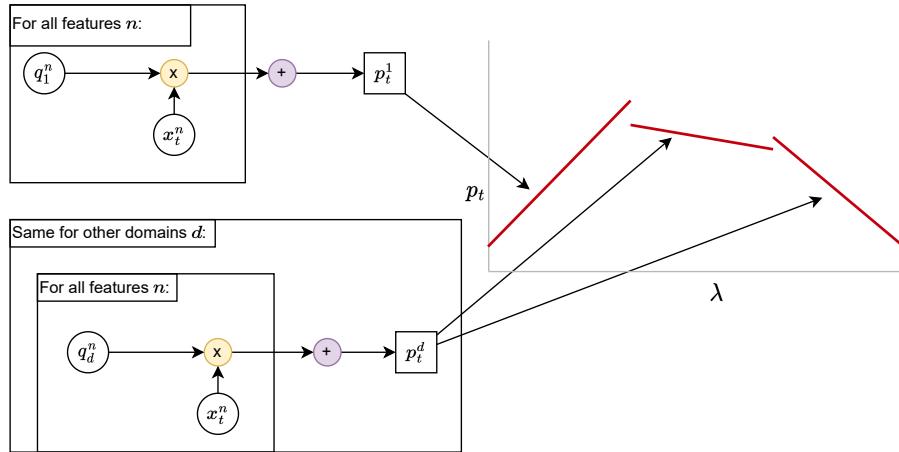
for each hour  $t \in T$  do
    Receive feature vector  $\mathbf{x}_t$ 
     $h_t = \text{mod}(t, 24)$ 
     $p_t^F(\lambda) = \alpha^F [\mathbf{q}_{h_t}^{F\top} \mathbf{x}_t]$ 
     $p_t^H(\lambda) = \alpha^H [\mathbf{q}_{h_t}^{H\top} \mathbf{x}_t]$ 
    Construct bidding curve from  $p_t^F(\lambda)$  and submit
    Cleared forward market determines  $p_t^F(\lambda_t^F)$  and  $p_t^H(\lambda_t^F)$ 
    Realize uncertainties and receive revenue from  $\mathbf{R}(p_t^F(\lambda_t^F), p_t^H(\lambda_t^F))$ 
end for

```

---

#### 5.5.3 GAPD

The structure of the *General* model including price domains is illustrated in figure 32. The schematic explicitly shows the parameter multiplication and addition for constructing the price curve for the first domain, and generalizes the process for the remaining domains.



**Figure 32:** Schematic overview of *General* architecture with price domains

The operation of the model is given in algorithm 10

---

#### Algorithm 10 GAPD

---

```

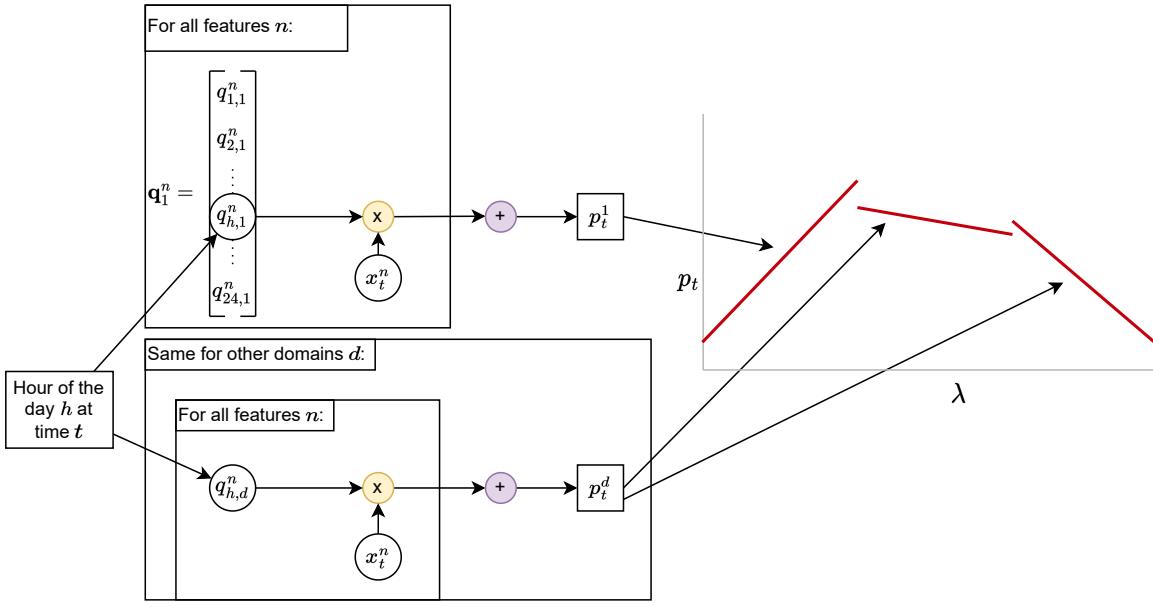
for each hour  $t \in T$  do
    Receive feature vector  $\mathbf{x}_t$ 
    for each domain  $d \in [1..3]$  do
         $p_{d,t}^F(\lambda) = \alpha^F [\mathbf{q}_d^{F\top} \mathbf{x}_t]$ 
         $p_{d,t}^H(\lambda) = \alpha^H [\mathbf{q}_d^{H\top} \mathbf{x}_t]$ 
    end for
    Construct bidding curve from  $[p_{1,t}^F(\lambda), p_{2,t}^F(\lambda), p_{3,t}^F(\lambda)]$  and submit
    Cleared forward market determines  $p_t^F(\lambda_t^F)$  and  $p_t^H(\lambda_t^F)$ 
    Realize uncertainties and receive revenue from  $\mathbf{R}(p_t^F(\lambda_t^F), p_t^H(\lambda_t^F))$ 
end for

```

---

#### 5.5.4 HAPD

The structure of the *Hourly* model including price domains is illustrated in figure 33. As in the schematic for **GAPD**, the construction for the first domain is shown explicitly, and generalized for the other domains.



**Figure 33:** Schematic overview of *Hourly* architecture with price domains

The operation of the model is given in algorithm 11.

---

#### Algorithm 11 HAPD

---

```

for each hour  $t \in T$  do
    Receive feature vector  $\mathbf{x}_t$ 
     $h_t = \text{mod}(t, 24)$ 
    for each domain  $d \in [1..3]$  do
         $p_{d,t}^F(\lambda) = \alpha^F \left[ \mathbf{q}_{h_t,d}^F \top \mathbf{x}_t \right]$ 
         $p_{d,t}^H(\lambda) = \alpha^H \left[ \mathbf{q}_{h_t,d}^H \top \mathbf{x}_t \right]$ 
    end for
    Construct bidding curve from  $[p_{1,t}^F(\lambda), p_{2,t}^F(\lambda), p_{3,t}^F(\lambda)]$  and submit
    Cleared forward market determines  $p_t^F(\lambda_t^F)$  and  $p_t^H(\lambda_t^F)$ 
    Realize uncertainties and receive revenue from  $\mathbf{R}(p_t^F(\lambda_t^F), p_t^H(\lambda_t^F))$ 
end for

```

---

#### 5.5.5 GAPDR

The retraining procedure is a higher-level functionality added on top of the model implementation, which dictates a functional relation between the training and testing context. The structure of model **GAPDR** is therefore identical to **GAPD**, and constructing an algorithm for **GAPDR** requires the training and testing context to be involved. Algorithm 12 shows how algorithm 10 is implemented in the retraining framework from the generic algorithm 6.

---

**Algorithm 12 GAPDR**

---

**Require:** Historical data  $\mathcal{T}$

**for** each month  $m \in T^m$  **do**

- Train parameters  $\mathbf{q}^F$  and  $\mathbf{q}^H$  by solving (81)-(96) on data  $\mathcal{T}$  using algorithm 10
- Run algorithm 10 with  $T = m$  to realize revenue
- Append  $|m|$  new datapoints to  $\mathcal{T}$
- Remove  $|m|$  oldest datapoints from  $\mathcal{T}$

**end for**

---

**5.5.6 HAPDR**

The same is true for **HAPDR** compared to **HAPD**, and so the algorithm is provided below without further explanation.

---

**Algorithm 13 HAPDR**

---

**Require:** Historical data  $\mathcal{T}$

**for** each month  $m \in T^m$  **do**

- Train parameters  $\mathbf{q}^F$  and  $\mathbf{q}^H$  by solving (81)-(96) on data  $\mathcal{T}$  using algorithm 11
- Run algorithm 11 with  $T = m$  to realize revenue
- Append  $|m|$  new datapoints to  $\mathcal{T}$
- Remove  $|m|$  oldest datapoints from  $\mathcal{T}$

**end for**

---

$$\mathbf{p}^F = \{p_t^F | \forall t \in T\} \quad (135)$$

$$\mathbf{q}_d^\top \mathbf{x}_t = \sum_{n=1}^N q_d^n x_t^n = q_d^\lambda \lambda + \sum_{n=2}^N q_d^n x_t^n \quad (136)$$

$$\mathbf{q}_d = \begin{bmatrix} q_d^\lambda \\ q_d^2 \\ \vdots \\ q_d^N \end{bmatrix} \quad (137)$$

## 6 Real Time Adjustment

Since the electrolyzer is not subject to ramping constraints, it is possible to perform real-time adjustment of the hydrogen production. That means the electrolyzer can be used to mediate deviations from the production that was committed in the forward market - either by turning down the electrolyzer in times of deficit production, or turning up the electrolyzer in times of surplus production. However, the balancing prices might be such that a surplus deviation would rather be incurred in the balancing stage than directing the excess to the electrolyzer. Conversely, it might sometimes be more profitable to incur a deficit settlement in the balancing stage, than turning down the electrolyzer. This chapter will investigate this concept of real-time adjustment, and develop different strategies for utilizing it. In reality, the balancing price is not settled until after the hour has passed, and is therefore technically unknown at the time of adjustment. However, very good predictions can be obtained within the hour by considering the previous system status<sup>44</sup> as well as the trades observed in the intra-day market leading up to the hour. For this reason, the balancing price is assumed to be known at the time of adjustment, instead of modelling it as a very accurate forecast.

The daily production requirement results in the producer having to operate the electrolyzer to produce hydrogen, even in hours when the most profitable operation for that hour might be not to produce. The producer has to decide when to produce hydrogen, before knowing the balancing prices<sup>45</sup>, to ensure the daily requirement is met. This production schedule is exactly what the deterministic and feature-driven models are outputting, because they utilize data available the day before to output the production schedule for each hour of the following day, subject to the minimum daily production constraint. The adjustment strategies developed in this chapter are completely independent of how this schedule was originally constructed, and can therefore be implemented generally to any hydrogen production schedule.

### 6.1 Adjustment Decision Rule

Looking at a single hour and without considering the minimum daily hydrogen production requirement, the producer is faced with the decision of how to operate the electrolyzer within its capacity defined by  $H$ , given a realized production and the balancing prices, such as to maximize the revenue from the balancing settlements and hydrogen production. The preferred amount of hydrogen production thus maximizes the combined revenue from balancing settlements and hydrogen production, as given by equation (138).

$$p_t^{H*} = \underset{p_t^H \in H}{\operatorname{argmax}} \left[ \lambda_t^{DW} (E_t^{real} - p_t^F - p_t^H)^+ - \lambda_t^{UP} (p_t^F + p_t^H - E_t^{real})^+ + \lambda_t^H p_t^H \right] \quad (138)$$

$$H = \left\{ p_t^H : 0 \leq p_t^H \leq \bar{P}^H \right\} \quad (139)$$

Since the forward market has already been cleared,  $p_t^F$  is no longer a variable. The power production  $E_t^{real}$  has also been realized, meaning the producer can adjust their deviations in the balancing stage directly by adjusting  $p_t^H$ . The result to be derived is thus a procedure for deciding the value of  $p_t^H$  in real-time.

Recall the rules of the dual pricing scheme from equations (15)-(17):

$$\lambda_t^{DW} \leq \lambda_t^F \quad \forall t \in T \quad (15)$$

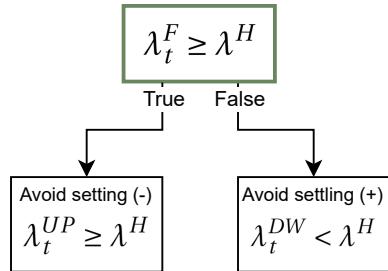
$$\lambda_t^{UP} \geq \lambda_t^F \quad \forall t \in T \quad (16)$$

$$(\lambda_t^{UP} - \lambda_t^F)(\lambda_t^{DW} - \lambda_t^F) = 0 \quad \forall t \in T \quad (17)$$

<sup>44</sup>The system status one hour ahead is predicted with 92% accuracy in [15].

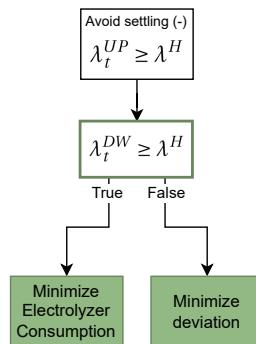
<sup>45</sup>And before having an accurate forecast of them.

When the production is realized, the forward price is known, since it was cleared during the day before. If the forward price is higher than the hydrogen price, by equation (16) it would be better to turn down hydrogen production, than to settle a deficit in the balancing stage, resulting in a net gain of  $(\lambda^{UP} - \lambda^H)$  per MW of deficit. Conversely, if the forward price is lower than the hydrogen price, by equation (15) it would be more profitable to direct any excess production into the electrolyzer, than settling it in the balancing stage, resulting in a net gain of  $(\lambda^H - \lambda^{UP})$  per MW of excess production. The following branching can thus be established, where (+) indicates a realized excess, and (-) indicates a realized deficit:



**Figure 34:** Forward price branching

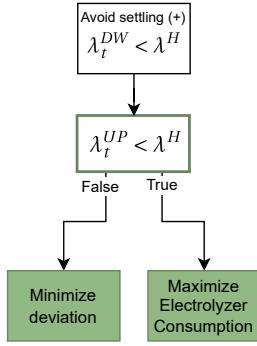
Let us first consider the case where  $\lambda_t^F \geq \lambda^H$ , and consequently  $\lambda_t^{UP} \geq \lambda^H$ , where it is certain that settling a deficit should be avoided. If the balancing price for surplus deviation  $\lambda_t^{DW}$  is also greater than  $\lambda^H$ , it would be more profitable to settle a surplus production, than to direct it into the electrolyzer. If  $\lambda_t^{DW}$  is less than  $\lambda^H$ , by figure 34 it is already concluded that settling a surplus should be avoided, meaning any deviation from the forward bid should be minimized. This reasoning is summarized in figure 35:



**Figure 35:** High forward price scenario

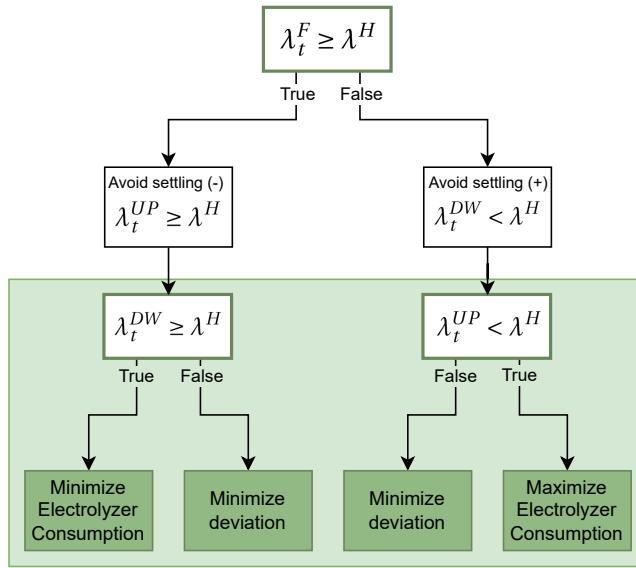
Let us now consider the other case, where  $\lambda_t^F < \lambda^H$ , and consequently  $\lambda_t^{DW} < \lambda^H$ , where it is certain that settling a surplus should be avoided.

If the balancing price for deficit deviation  $\lambda_t^{UP}$  is also less than  $\lambda^H$ , the electrolyzer should be adjusted to maximum capacity, since there is a net gain by incurring a deficit in the balancing stage and directing it to the electrolyzer. If  $\lambda_t^{UP}$  is greater than  $\lambda^H$ , by figure 34 it is already concluded that settling a deficit should be avoided, meaning any deviation from the forward bid should be minimized. This reasoning is illustrated in figure 36:



**Figure 36:** Low forward price scenario

Putting these sections together, a complete decision rule can be constructed, where the optimal decision is given by the value of the balancing prices:



**Figure 37:** Complete decision rule

Given the balancing prices<sup>46</sup>, the optimal electrolyzer operation for a given isolated hour is given directly by figure 37. The remaining part of this section will derive the mathematical formulation of this decision rule, and prove that only the bottom area marked with a green background is actually required for a complete formulation.

In order to utilize figure 37 in an adjustment algorithm, the surplus or deficit power after meeting the forward bid  $d_t$  is calculated, and the result from the decision rule is denoted by the function  $\Pi$ :

$$d_t = E_t^{real} - p_t^F \quad (140)$$

$$p_t^{H*} = \Pi(d_t, \lambda_t^F, \lambda_t^{UP}, \lambda_t^{DW}) \quad (141)$$

Note that if a model has allocated any production to the electrolyzer,  $d_t$  will be equal to this amount plus the difference between the forecasted and realized production. Also note that the function  $\Pi$  should not

<sup>46</sup>Or a very accurate forecast of them.

---

return the adjustment (change in value) of the schedule, but the actual appropriate value of the hydrogen production in the range defined by  $H$  in equation (139).

Repeating equations (15) and (16) again, the following relation follows:

$$\lambda_t^{DW} \leq \lambda_t^F \quad \forall t \in T \quad (15)$$

$$\lambda_t^{UP} \geq \lambda_t^F \quad \forall t \in T \quad (16)$$

↓

$$\lambda_t^{DW} \leq \lambda_t^{UP} \quad \forall t \in T \quad (142)$$

Meaning any upper bound on  $\lambda_t^{UP}$  is also an upper bound on  $\lambda_t^{DW}$ , and any lower bound on  $\lambda_t^{DW}$  is also a lower bound on  $\lambda_t^{UP}$ . This serves the following implications:

$$\lambda_t^{DW} \geq \lambda^H \Rightarrow \lambda_t^{UP} \geq \lambda^H \quad (143)$$

$$\lambda_t^{UP} < \lambda^H \Rightarrow \lambda_t^{DW} < \lambda^H \quad (144)$$

Meaning that the true-condition on either of the bottom two green-bordered conditional cases in figure 37 uniquely identifies an optimal result. If  $\lambda_t^{DW} \geq \lambda^H$  it is optimal to minimize electrolyzer consumption, and if  $\lambda_t^{UP} < \lambda^H$  it is optimal to maximize electrolyzer consumption, with both of these cases being mutually exclusive. To uniquely identify the two remaining paths, a further condition on the forward price is needed, but since they both contain the same results, to minimize deviation, this can be caught by an else-condition. The function  $\Pi$  can thus be simplified to the following closed form expression:

$$\Pi(d_t, \lambda_t^{UP}, \lambda_t^{DW}) = \begin{cases} 0, & \lambda_t^{DW} > \lambda^H \\ \bar{P}_t^H, & \lambda_t^{UP} < \lambda^H \\ \alpha^H[d_t], & \text{else} \end{cases} \quad (145)$$

Where  $\alpha^H[\cdot]$  is the feasibility bound for the electrolyzer given by equation (113).

If there was no daily production requirement, this would conclude the electrolyzer modelling, and  $p_t^H$  would be irrelevant to output from the models from chapter 5, since no scheduling would be necessary<sup>47</sup>. Because there is a daily production requirement, the adjustment of a given schedule becomes more complicated. The following section investigates how the optimal decision rule developed in the present section can be utilized in different ways, while still respecting the daily production requirement.

## 6.2 Adjustment Policies

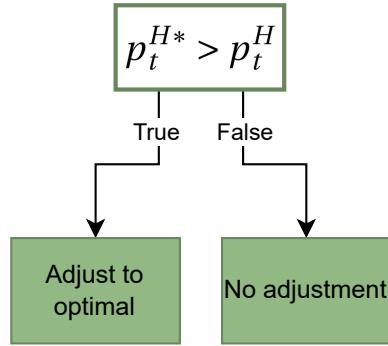
Since the independent optimal adjustment each hour is given by the decision rule  $\Pi(d_t, \lambda_t^{UP}, \lambda_t^{DW})$ , the goal of any adjustment policy is to come as close as possible to the behavior determined by this rule, constrained with different levels of complexity to guarantee the production requirement is met.

### 6.2.1 Adjusting Upwards

The simplest way to adjust the schedule while guaranteeing the requirement is met, is by only adjusting when the decision rule says to adjust upwards, illustrated in figure 38.

---

<sup>47</sup>Any inter-temporal production constraint would make scheduling necessary.



**Figure 38:** Upwards Adjustment

The algorithm for this adjustment policy then becomes:

---

**Algorithm 14** Upwards adjustment

---

**for** each hour  $t$  **do**

    Receive model output:  $p_t^F, p_t^H$

    Receive balancing price (forecasts):  $\lambda_t^{UP}, \lambda_t^{DW}$

    Realize production:  $E_t^{real}$

    Compute real-time deviation  $d_t = E_t^{real} - p_t^F$

    Compute optimal hydrogen production  $p_t^{H*} = \Pi(d_t, \lambda_t^{UP}, \lambda_t^{DW})$

**if**  $p_t^{H*} > p_t^H$  **then**

$p_t^{adj} = p_t^{H*}$

**else**

$p_t^{adj} = p_t^H$

**end if**

    Receive revenue from  $\mathbf{R}(p_t^F, p_t^{adj})$

**end for**

---

### 6.2.2 Adjusting Downwards

The next level of complexity introduces the possibility of adjusting down, and producing less than what was scheduled. A simple way of implementing this possibility while still guaranteeing to meet the requirement, is by allowing the model to adjust down only such that the sum of what has already been produced, and what is still scheduled to be produced, is at least equal to the required amount.

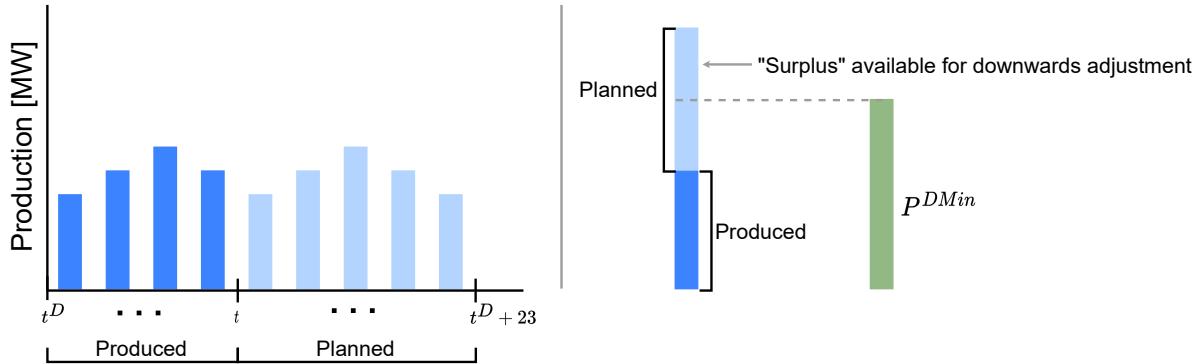
The scenario of having surplus hydrogen production could happen either if there actually is a profit to be gained from scheduling more than the requirement in the given day, or if the model did some upwards adjustment in earlier hours, or if there was scheduled more than the required amount due to the limitations of the model<sup>48</sup>.

In the implementation of the downwards adjustment algorithm, some new variables are introduced, such that the process can be divided into several smaller steps. Since the algorithm must now consider what has been done in earlier hours of the same day, and what is scheduled for later hours, the algorithm operates with the daily time index  $t^D \in T^D$ , which indicates the first index of a day meaning the entire day is indexed

---

<sup>48</sup>The deterministic model might sub-optimally schedule more than the required amount due to an inaccurate price forecast. The feature-driven model is not solving an optimization problem in the bidding phase and is therefore not expected to produce the exact required amount for unseen feature vectors.

by  $t \in [t^D, t^D + 23]$ , as described in detail in subsection 5.1.2. Figure 39 illustrates the concept, and serves as a reference for the introduced variables.



**Figure 39:** Produced and planned hydrogen production

The first variable that will be introduced is  $p_t^{produced}$ , which is how much hydrogen that has already been produced in the given day. This variable is initialized to zero at the beginning of each day, and incremented with the resulting  $p_t^{adj}$  at the end of each hour, thus being a running total of the produced hydrogen.

$$p_{t^D}^{produced} = 0 \quad \forall t^D \in T^D \quad (146)$$

$$p_{t+1}^{produced} = p_t^{produced} + p_t^{adj} \quad \forall t \in [t^D..t^D + 22], \quad \forall t^D \in T^D \quad (147)$$

$p_t^{planned}$  is the amount scheduled for the remaining hours, including the present hour:

$$p_t^{planned} = \sum_{i=t}^{t^D+23} p_i^H \quad \forall t \in [t^D..t^D + 23], \quad \forall t^D \in T^D \quad (148)$$

From these two quantities, along with the daily production requirement  $p^{DMin}$ , the variable  $p_t^{surplus}$  can be computed, which is the amount available for downwards adjustment that still ensures the daily requirement is met:

$$p_t^{surplus} = p_t^{produced} + p_t^{planned} - p^{DMin} \quad \forall t \in [t^D..t^D + 23], \quad \forall t^D \in T^D \quad (149)$$

The final variable that will be introduced is  $p_t^{wanted}$ , which is the amount of downward adjustment that would be preferred according to the decision rule:

$$p_t^{wanted} = p_t^H - p_t^{H*} \quad \forall t \in [t^D..t^D + 23], \quad \forall t^D \in T^D \quad (150)$$

Being a running total,  $p_t^{produced}$  will be included in all iterations, but the remaining variables that have been introduced are only relevant if the optimal adjustment is downwards, and is therefore only computed in the case of  $p_t^{H*} \leq p_t^H$ .

The upwards and downwards adjustment algorithm will thus be an augmented version of algorithm 14, where the case of  $p_t^{H*} \leq p_t^H$  is expanded to check for feasible downwards adjustment<sup>49</sup>. If there is enough surplus available, meaning  $p_t^{surplus} \geq p_t^{wanted}$ , then the optimal adjustment can be used directly.

<sup>49</sup>The case is purposely *not* that  $p_t^{H*}$  is strictly less than  $p_t^H$ , even though that would reduce the amount of computations for the case of  $p_t^{H*} = p_t^H$ . This nuance will guarantee upwards adjustment in the case of the scheduled production not meeting the daily requirement, as explained in detail in the final paragraph of the present subsection.

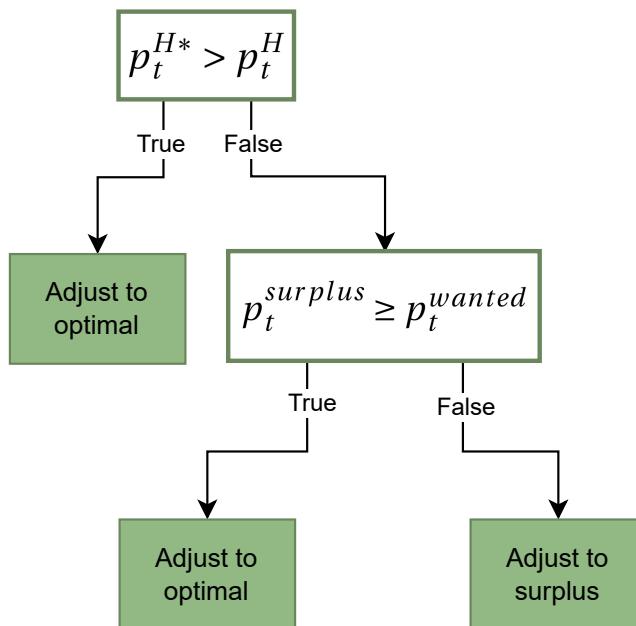
---


$$p_t^{adj} = p_t^{H*}, \quad \text{if } p_t^{surplus} \geq p_t^{wanted} \quad (151)$$

Otherwise, the resulting production will be set to the scheduled production with the surplus subtracted, such that at all the available downward adjustment is utilized, even though the optimal is not reached<sup>50</sup>:

$$p_t^{adj} = p_t^H - p_t^{surplus}, \quad \text{if } p_t^{surplus} < p_t^{wanted} \quad (152)$$

The upwards and downwards adjustment policy is described fully by algorithm 15, and is illustrated in figure 40.



**Figure 40:** Upwards and Downwards Adjustment

---

<sup>50</sup>It might seem necessary to apply the  $(\cdot)^+$  operator to ensure the requirement  $p_t^{adj} \geq 0$ , but this is already guaranteed. See section A.4 in the Appendix for the proof of this statement.

---

**Algorithm 15** Upwards and downwards adjustment
 

---

```

for each day  $t^D \in T^D$  do
  Initialize  $p_t^{produced} = 0$ 
  for each hour  $t \in [t^D, t^D + 23]$  do
    Receive model output:  $p_t^F, p_t^H$ 
    Receive balancing price (forecasts):  $\lambda_t^{UP}, \lambda_t^{DW}$ 
    Realize production:  $E_t^{real}$ 
    Compute real-time deviation  $d_t = E_t^{real} - p_t^F$ 
    Compute optimal hydrogen production  $p_t^{H*} = \Pi(d_t, \lambda_t^{UP}, \lambda_t^{DW})$ 
    if  $p_t^{H*} > p_t^H$  then
       $p_t^{adj} = p_t^{H*}$ 
    else
       $p_t^{planned} = \sum_{i=t}^{t^D+23} p_i^H$ 
       $p_t^{surplus} = p_t^{produced} + p_t^{planned} - p^{DMin}$ 
       $p_t^{wanted} = p_t^H - p_t^{H*}$ 
      if  $p_t^{surplus} \geq p_t^{wanted}$  then
         $p_t^{adj} = p_t^{H*}$ 
      else
         $p_t^{adj} = p_t^H - p_t^{surplus}$ 
      end if
    end if
    Receive revenue from  $\mathbf{R}(p_t^F, p_t^{adj})$ 
     $p_{t+1}^{produced} = p_t^{produced} + p_t^{adj}$ 
  end for
end for

```

---

It might seem like the computation of  $p_t^{surplus}$  should be computed as:

$$p_t^{surplus} = (p_t^{produced} + p_t^{planned} - p^{DMin})^+ \quad (153)$$

Such that the variable will take a value of zero if downward adjustment is unavailable. However, by allowing  $p_t^{surplus}$  to take negative values, the resulting computation of the adjustment will correct for any mistake in the scheduling, that caused the daily production requirement not to be met<sup>51</sup>. It is not guaranteed that all missing production will be mediated, but it is guaranteed that the schedule of every hour will be adjusted upwards with a strictly positive amount until it is. This upwards adjustment is guaranteed, because in the case of such a scheduling mistake (which results in  $p_t^{surplus} < 0$ ), one of the two following to scenarios must occur:

1. Either the production is turned up by the case of  $p_t^{H*} > p_t^H$ .
2. Or the condition of  $p_t^{surplus} \geq p_t^{wanted}$  will necessarily be false, since  $p_t^{surplus} < 0$ , and  $p_t^{wanted}$  must be non-negative since this block is already conditioned on  $p_t^{H*} \leq p_t^H$ . This causes the resulting hydrogen production to be set as  $p_t^H - p_t^{surplus}$ , and since  $p_t^{surplus} < 0$ , it is given that  $p_t^{adj} > p_t^H$ , meaning there is upwards adjustment.

In any real-world application, the algorithm will thereby mediate practically all cases of the requirement not being met, but in order to provide a theoretical guarantee, a check would have to be implemented in the block with the  $p_t^{H*} > p_t^H$  conditional.

---

<sup>51</sup>Which can occur only for the feature-driven models.

---

### 6.2.3 Postponing Production

Yet another and more complex adjustment algorithm was developed in the project, where postponing of the production to later hours was included as a possibility. This approach was not guaranteed to have a non-negative impact on the performance<sup>52</sup>, since the forward price for later hours<sup>53</sup> was used as a proxy for the balancing price, and production was then postponed if the forward price indicated that it would result in a net gain. Using the forward price in this way introduced a risk of incurring a loss if the balancing price was realized at a different value than the forward price. This mechanism was added on top of algorithm 15, and activated in the case that  $p_t^{H*} < p_t^H$  and without any surplus available. The algorithm presented such complexity, that significant gains from implementing the algorithm was required to justify including the algorithm in the report. The postponing algorithm provided a slight increase in performance in some of the early iterations of learned models, but for the final models presented in section 5.5, the postponing algorithm resulted in *worse* performance than algorithm 15. The postponing algorithm does not constitute a significant contribution to the project, and is therefore only included for reference and as inspiration for future work in section A.5 in the Appendix.

---

<sup>52</sup>The two presented algorithms *are* guaranteed to have a non-negative impact on the performance, given the assumption that the balancing prices are known (or that the forecasts are perfect).

<sup>53</sup>Which is known for all hours in the day at the time of adjustment.

---

## 7 Benchmarks for Evaluation

The deterministic model and different feature-driven models will be compared against each other, but other insights can be gained from developing additional benchmarks to evaluate the models against. In this section, a variety of upper bounds to compare the models against are developed. All models are evaluated using the final revenue function given in equation (63), as seen in all the algorithms developed in the project.

### 7.1 Hindsight

By solving the base model in (49)-(62) for the test period directly with all uncertainties known, the operation can be calculated in "hindsight". This will result in the highest possible revenue being generated. This enables the assessment of how much further improvement is theoretically possible on the models, and results in the fairly simple algorithm:

---

#### Algorithm 16 Hindsight

Reveal all uncertain parameters  $E_t^{real}$ ,  $\lambda_t^{UP}$  and  $\lambda_t^{DW} \forall t \in T$

Solve base model (49)-(62) to get  $\mathbf{p}^F$  and  $\mathbf{p}^H$  for entire period

Receive revenue from  $\mathbf{R}(\mathbf{p}^F, \mathbf{p}^H)$

---

This model constitutes the absolute upper bound on any model. It is practically unachievable by any model, and thus serves as a theoretical maximum, providing a perspective on how well the different models and algorithms perform, and how much potential exists for improvement.

### 7.2 Optimal Adjustment

By computing the forward bids of a model for the entire test period, and then running algorithm 16 with these forward bids as fixed parameters, the optimal hydrogen production associated with these bids can be found. This can be thought of as a hindsight adjustment algorithm, that is allowed to adjust the hydrogen production with all uncertainties revealed, but with fixed forward bids. The resulting performance can thus serve as a theoretical upper bound on the performance of any adjustment algorithm on a given model. The optimal adjustment benchmark is therefore specific to a given model, since it requires the forward bids of a specific model as input. The algorithm for constructing the **Optimal Adjustment** benchmark is given in algorithm 17:

---

#### Algorithm 17 OA

**Require:**  $\mathbf{p}^F$  from a given model for entire test period

Reveal all uncertain parameters  $E_t^{real}$ ,  $\lambda_t^F$ ,  $\lambda_t^{UP}$  and  $\lambda_t^{DW} \forall t \in T$

Solve base model (49)-(62) with  $\mathbf{p}^F$  fixed to get  $\mathbf{p}^H$  for entire period

Receive revenue from  $\mathbf{R}(\mathbf{p}^F, \mathbf{p}^H)$

---

### 7.3 Static Oracle

Another benchmark that can provide valuable information is a feature-driven model that is trained on the test set instead of the training set. This model is thus not a separate algorithm, but any of the algorithms from section 5.5 trained using the test set as the historical data. Each type of feature-driven model<sup>54</sup> thus has a corresponding static oracle model, that provides an upper bound on how well the feature-driven model can perform on the test set. The benchmark is called a *static* oracle, because the model is trained using the

---

<sup>54</sup>Along with each of the feature vectors.

---

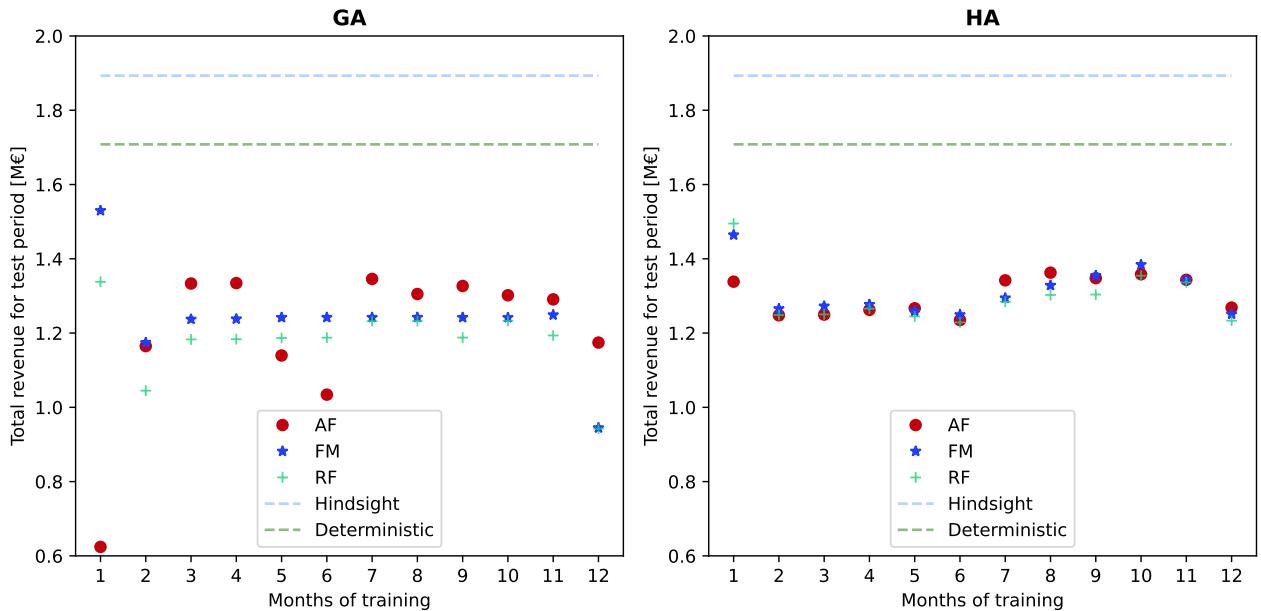
entire test set as the training period, meaning no retraining or varying of training period is performed. The static oracle benchmark will be used on few selected feature-driven models when deemed relevant.

## 8 Results

In this section, the total revenue obtained by the different algorithms in the test period is used to compare the models against each other. The revenue generated by the models is given by the Revenue function from equation (63) in subsection 5.1.2. The test period consists of the entire year of 2020.

### 8.1 Feature-driven Models

Figure 41 shows an overview of the results of models **GA**<sup>55</sup> and **HA**<sup>56</sup>, with the total revenue for the test period given as a function of training period length. Each type of feature vector is indicated with a different marker on the plot, where red circles indicate All Features, blue stars indicate the Forecast Model feature vector, and the green plusses indicate the Reduced Feature vector. The revenue generated by the deterministic model from algorithm 2 and the upper bound given by the hindsight model from algorithm 16 are indicated as well.



**Figure 41:** Results for models **GA** and **HA**

Models **GA** and **HA** clearly show very poor performance, with none of the models outperforming the deterministic model. Applying price domains to the models improves the performance significantly, as can be seen on figure 42, where the results for models **GAPD**<sup>57</sup>, **HAPD**<sup>58</sup>, **GAPDR**<sup>59</sup> and **HAPDR**<sup>60</sup> are shown<sup>61</sup>.

<sup>55</sup>Algorithm 8.

<sup>56</sup>Algorithm 9.

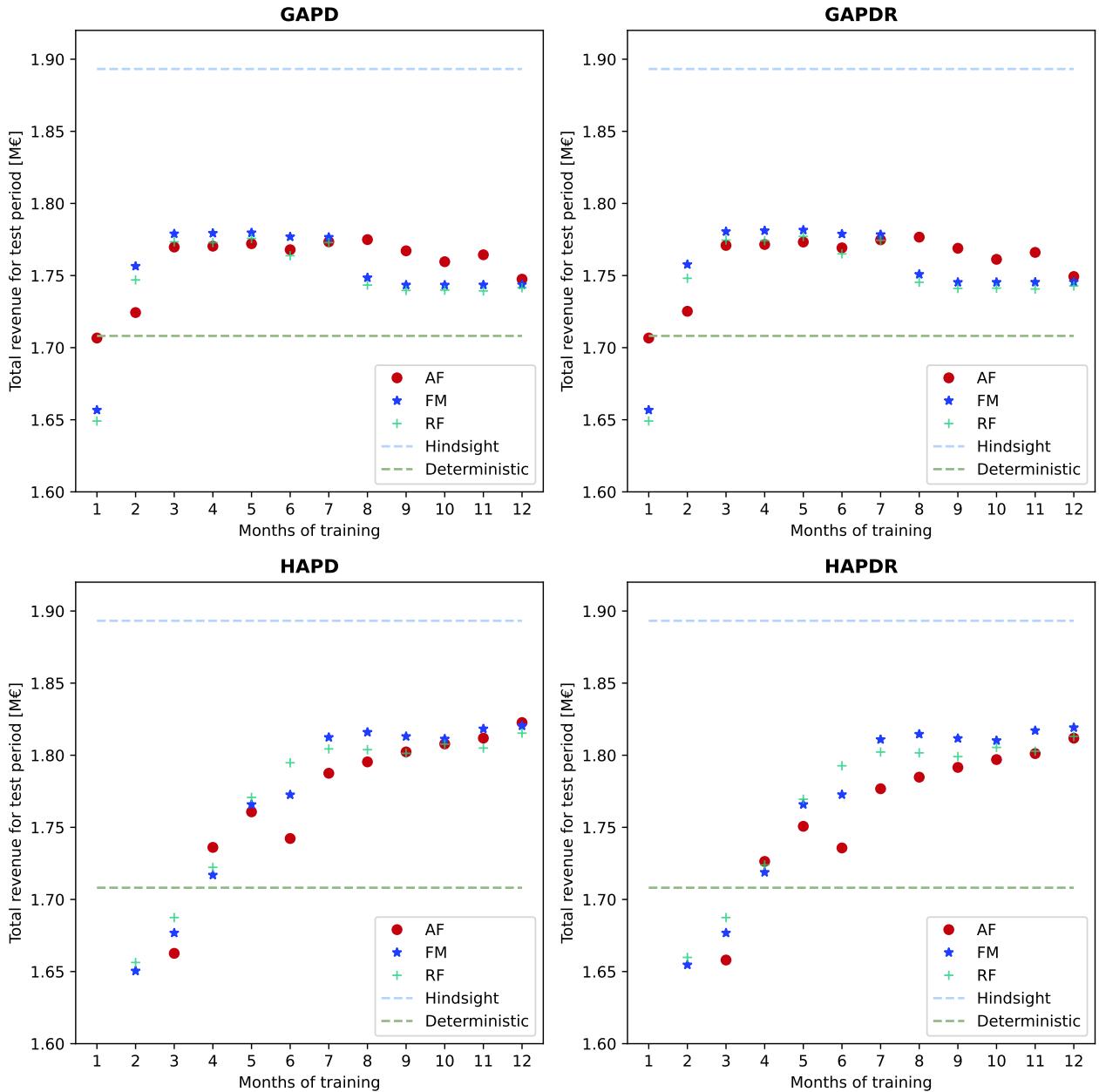
<sup>57</sup>Algorithm 10.

<sup>58</sup>Algorithm 11.

<sup>59</sup>Algorithm 12.

<sup>60</sup>Algorithm 13.

<sup>61</sup>An overview of all the results combined is found in section A.6 in the Appendix in a larger format to make it suitable for printing.



**Figure 42:** Results for models **GAPD**, **HAPD**, **GAPDR** and **HAPDR**

Looking at the **GAPD** model, it is clear that introducing price domains improves the results and causes more stable performance across different feature vectors as well as different training periods. The **GAPD** model outperforms the deterministic model given a training period of two months or more, with the **GAPD-FM** model reaching a slightly higher revenue than the other feature-vectors. Applying retraining in model **GAPDR** has basically no effect, and the final generated revenue reaches the same levels<sup>62</sup> as **GAPD**. Although **HAPD** and **HAPDR** show very high performance, the **GAPD** and **GAPDR** models actually perform significantly better on short training periods below 6 months, and overall show a more stable performance across different training periods<sup>63</sup>.

<sup>62</sup>All results differ with less than 0.2 % between the two models.

<sup>63</sup>The vertical axis is scaled to highlight the area of interest, resulting in the worst performing models not being visible on the plot. A figure with all models visible can be found in figure 56 in section A.6 in the Appendix for reference.

---

The **HAPD** model shows the best performance of all the models. An interesting result is that the performance of both **HAPD** and **HAPDR** tends to increase with longer training periods, which was not the trend seen for the *General* architecture. Both models seem to plateau around 7 months of training, where the trend of improvement continues afterwards but to a lesser degree.

Another interesting result that is not immediately obvious, is the difference in performance for different feature vectors for these models. The **RF** vector outperforms the **FM** vector up until 7 months of training, where the **FM** vector starts performing better. Since the general performance of both **HAPD** and **HAPDR** is fairly poor and unstable until this point, the results for more than 7 months of training are more relevant for the case of actually applying the model. Combining the forecast from SG with the aggregated forecasts from Energinet in a separate forecast model thus seems to provide a slight improvement in performance compared to only using the forecast from SG. It was expected that the decoupling of price and production training periods that was achieved using the **FM** vector would result in increased performance specifically for short training periods, which is contrary to what is observed. The reason for expecting better performance resulting from this decoupling was that the non-stationarity of the price data could then be captured, without sacrificing the longer term dependency for the production data. An explanation for why the **FM** vector does not excel for short training periods could be that the price data is in fact fairly stationary in the considered period. The data analysis from section 3.1.2 showed that there was some difference in the distributions of price days between 2019 and 2020, but it was significantly less than between the years 2020 and 2021, where the effect of using a forecast model might be more prevalent. Another reason could be that even though the forecast is improved using a forecast model that considers a long period of historical data, the value-oriented model itself might also require a longer training period to adjust to the improved forecast.

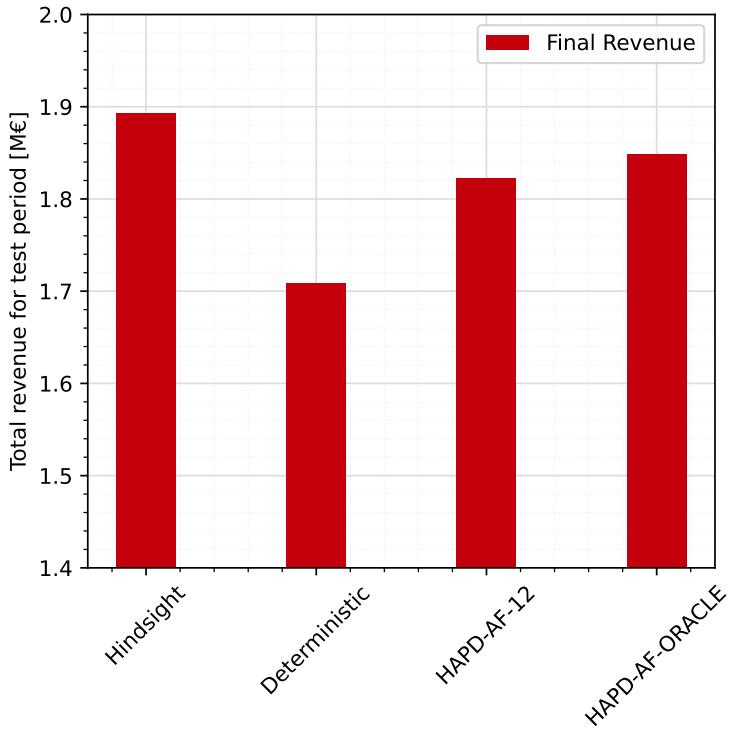
The **AF** vector reaches the same levels of performance as the other vectors, but evidently requires a longer training period to do so. The trend for the **AF** vector does not seem to plateau around 7 months of training as the others do, so it could be possible that the **AF** vector performs significantly better than the others, if given even longer training periods. As mentioned in section 5.3, introducing more parameters means more training data is required to generalize the model output, so it is completely expected that the **AF** vector requires longer training periods to reach the same performance as the other feature vectors, since it introduces significantly more parameters.

Based on these results, it is possible to conclude that including the aggregated forecasts from Energinet seems to provide a slight increase in performance for training periods longer than 7 months, and that combining the forecasts using a separate forecast model results in the model requiring shorter training periods to reach peak performance.

### 8.1.1 Evaluating HAPD

Model **HAPD** reaches the highest amounts of revenue. An interesting result is that applying retraining to the model does not show any improvement in performance. This is not a huge surprise, since retraining is especially suited for non-stationary environments where the most recent data is essential for good performance. The price data was shown to have a fairly small degree of non-stationarity in the data analysis in subsection 3.1.2 and the production data likewise in the introduction of section 3.2, and retraining was therefore not expected to have a large effect on performance. However, using retraining as such should not be disregarded due to this result, since it could very well provide a significantly increased performance in more non-stationary environments.

Figure 43 shows the revenue of the best performing version of **HAPD**, **HAPD-AF-12**, next to the corresponding static oracle, along with the hindsight and deterministic for comparison.



**Figure 43: HAPD-AF-12 with benchmarks**

The static oracle is only 1.4 % better than the trained model, meaning that the full potential is almost reached. For stationary data, the static oracle serves as a theoretical maximum of performance<sup>64</sup>, and the full 1.4 % improvement can therefore not be expected to be achieved, although some improvement might still be gained by increasing the training period, based on the trend in the results.

**HAPD-AF-12** achieves 6.70 % higher revenue than the deterministic model, and the hindsight model achieves only 3.87 % higher revenue than **HAPD-AF-12**.

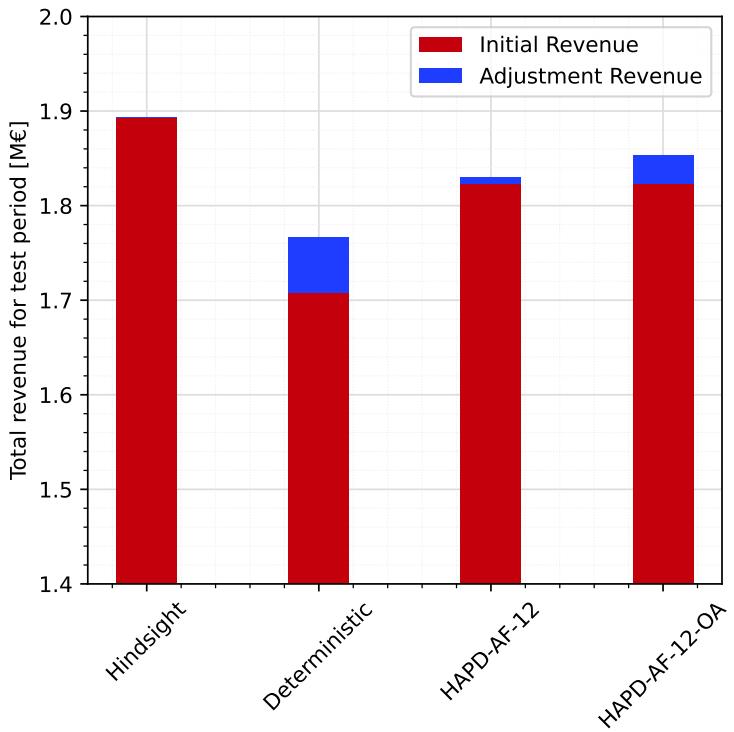
## 8.2 Adjustment

### 8.2.1 Upwards Adjustment

Figure 44 shows the revenue gain by applying upwards adjustment using algorithm 14 to both the learned and deterministic model. Along with the hindsight benchmark, the **Optimal Adjustment** benchmark from algorithm 17 is shown as **HAPD-AF-12-OA** as well, which serves as a theoretical maximum for the revenue gain obtained by applying adjustment, as described in subsection 7.2.

---

<sup>64</sup>If the data was highly non-stationary, a retraining model could potentially outperform a static oracle. Since the model considered here is not retrained, the static oracle does serve as a theoretical maximum.



**Figure 44:** Results of upwards adjustment algorithm

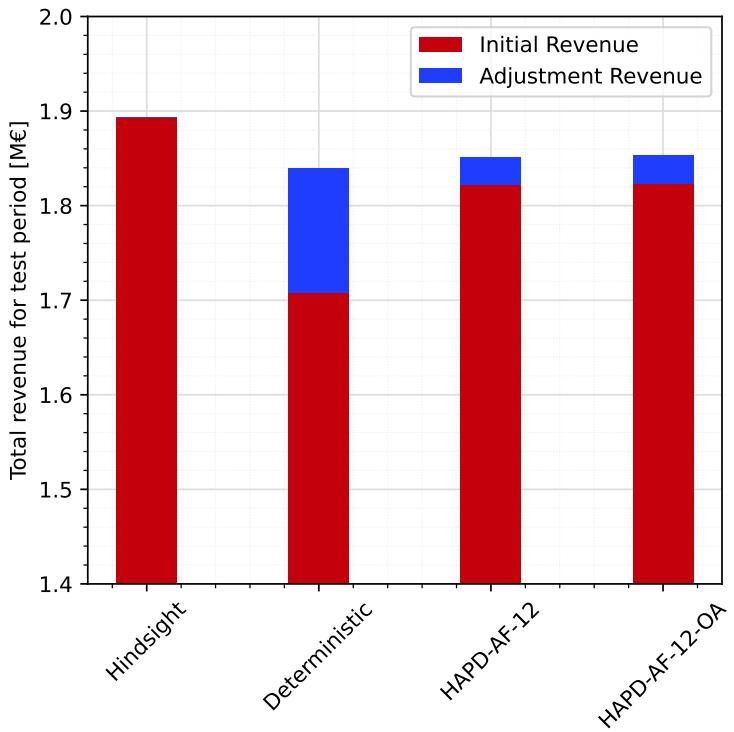
Upwards adjustment provides a revenue increase of 0.39 % (7,177 €) for the learned model, and 3.44 % (58,756 €) for the deterministic. Upwards adjustment thus seems to have diminishing returns the better the original performance of the model is, which is as expected. Comparing with the optimal adjustment benchmark reveals that the upwards adjustment algorithm 14 achieves 23.60 % of the theoretical maximum gain from adjustment for the **HAPD-AF-12** model.

To investigate why the adjustment algorithm has such a large effect on the deterministic model, the actual operation produced by the model is investigated. When investigating the adjustment of the deterministic model in detail, it is found that 83 % of the adjusted amount (79 % of the cases) are in times where the forward bid was not accepted due to an inaccurate price forecast<sup>65</sup>. Furthermore, only 47 % of the adjusted amount (48 % of the cases) was actually due to the electrolyzer being turned fully up, the rest being caused by minimizing the deviation between the realized amount and the amount sold in the forward market. In summary, this means that the large increase from upwards adjustment gained by the deterministic model is primarily explained by the inaccuracy of the price forecast causing the bid not to be accepted, and the algorithm simply directing the power to the electrolyzer that would otherwise have been sold in the forward market if the bid had been accepted. The implications of this will be further evaluated in section 8.3.

### 8.2.2 Upwards and Downwards Adjustment

Figure 45 shows the revenue gain by applying upwards and downwards adjustment using algorithm 15 to both the learned and deterministic model.

<sup>65</sup>Because the forecasting error was larger than the 5€/MW buffer.

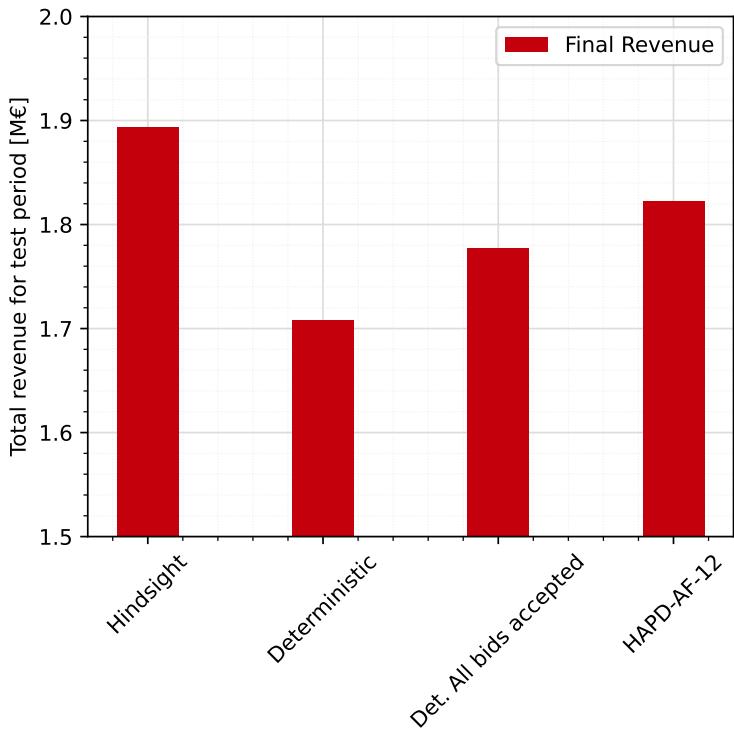


**Figure 45:** Results of upwards and downwards adjustment algorithm

Implementing algorithm 15 provides a total revenue increase of 1.58 % (28,711 €) for the learned model, and 7.73 % (131,969 €) for the deterministic. Comparing with the optimal adjustment benchmark reveals that algorithm 15 achieves 94.40 % of the theoretical maximum gain from adjustment for the **HAPD-AF-12** model. This is a surprisingly good result, and remembering the optimal adjustment is a *theoretical* maximum makes it even more significant.

### 8.3 Deterministic Model

The results from subsection 8.2.1 indicated that the deterministic model suffers a great loss from forward bids not being accepted, due to inaccurate price forecasts. The forward bids not being accepted will both have the consequence that production that would have been sold might be compensated with a lower price than in the forward market, and that consumption that would have been bought can cost the producer a higher amount when settling in the balancing stage than it would have if the bid had been accepted. An investigation shows that a total of 85,045 € is lost in potential revenue from the forward bids not being accepted, with 33 % of this amount being due to production bids not accepted, and the other 67 % being consumption bids. Figure 46 shows the performance of the deterministic model had all bids been accepted, along with the best performing learned model for reference.



**Figure 46:** Deterministic model with all bids accepted

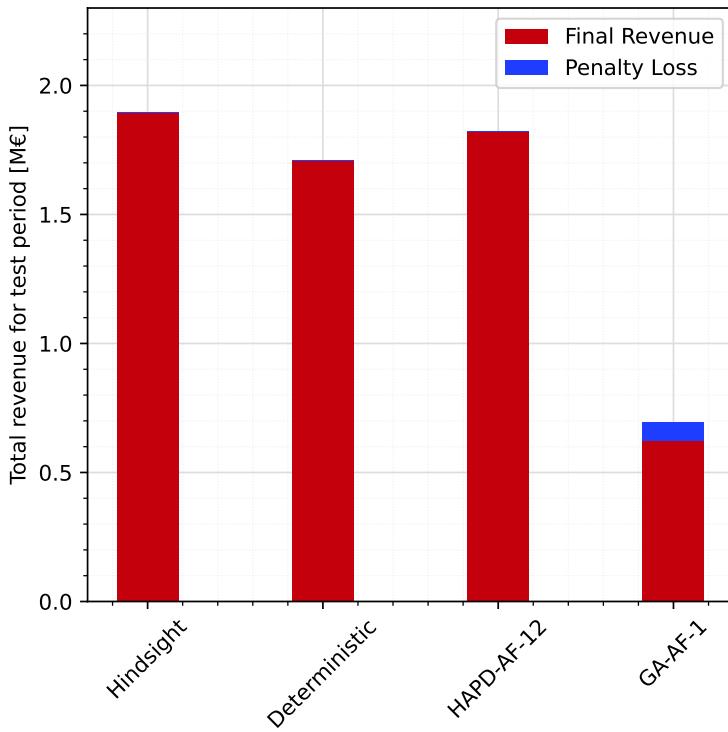
It is noted that the feature-driven model still outperforms the deterministic model with all bids accepted by 2.54 %.

The result can be interpreted as an estimate on the potential gain from changing the bidding strategy, where the result effectively corresponds to a maximum risk strategy that guarantees all bids are accepted<sup>66</sup>. The risk is incurred by placing the bids at less favorable thresholds, but a small investigation shows that appropriate thresholds might not necessarily incur that large a risk. The reasoning behind this statement is that production bids should be delivered at a minimum price of 0€/MW, since anything less would guarantee a loss for the producer, and the consumption bid should be delivered at a maximum price of  $\lambda^H$ , since any consumption bid higher than this would mean a net loss for the producer, since the hydrogen production has a fixed compensation of  $\lambda^H$ . Although this would be interesting to investigate further, such endeavors are left for future work.

## 8.4 Hydrogen Penalty

As described in section 4.3, a penalty of  $4.6 \frac{\text{€}}{\text{kg}_{\text{H}_2}}$  is incurred for missing the daily production requirement, corresponding to 80.61 € for each MWh of the required amount not allocated to the electrolyzer. The model that incurred the highest penalty of all was **GA-AF-1**, which is also the model with the lowest total score. By registering the missing production at the end of each day, the amount of production missing throughout the year (and thus the total amount being penalized) is found to be 866 MWh of missing allocation in total for the model. Figure 47 shows model **GA-AF-1** alongside the best performing learned model **HAPD-AF-12**, the deterministic model and the hindsight benchmark.

<sup>66</sup>A strictly theoretical guarantee is infeasible since it would require bidding production at a price of  $-\infty$  €/MW and consumption at a price of  $\infty$  €/MW, but practically translated to a feasible bid would mean bidding production at as low a price as allowed, and consumption at as high a price as allowed, which is potentially very sub-optimal, and thus a more risky approach.



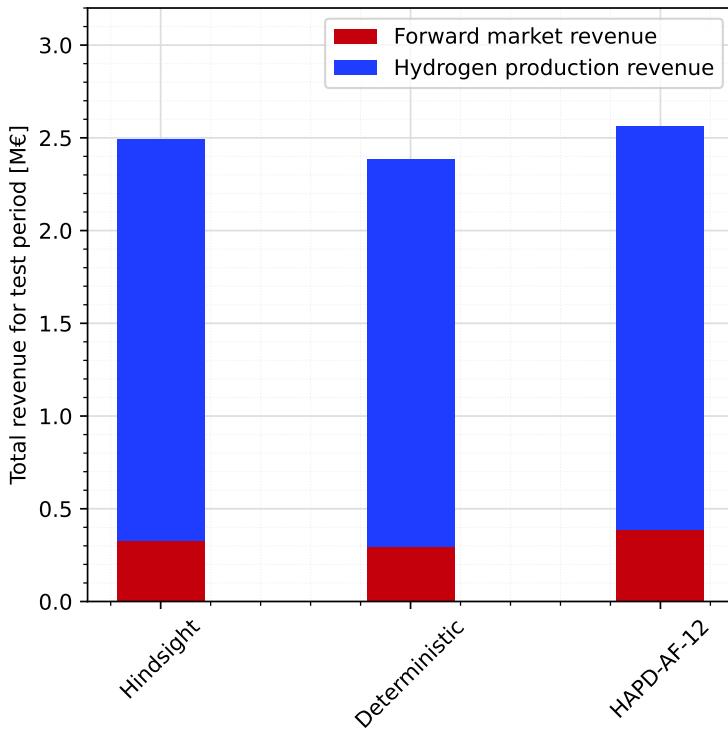
**Figure 47:** Impact of missing production penalty

It is noted that the penalized amount is insignificant compared to the total achieved revenue, and is thus not the reason for the poor performance of the model. This in turn means that the penalty is not the reason for the poor performance of any of the models, since **GA-AF-1** is the model being penalized the most. It is also noted that **HAPD-AF-12** never misses the daily production requirement. When investigating missing production for all the different models, it is found that the *General* architecture models trained for 2 months or more all have missing production in the order of 10-100 MWh throughout the year. All models with the *Hourly* architecture trained for 2 months or more always fulfill the daily production requirement and thus incur no penalty. All models with only 1 month of training have missing production in the order of 100-800 MWh across the year. The tendency that shorter training periods results in a larger amount of missing production is not a surprise, since longer training periods exposes the models to a more diverse selection of features, and thus enforces the parameter to fulfill the quota within a larger space of inputs. This result is particularly relevant if the models are to be applied to non-stationary data where shorter training periods are generally preferred, and extra measures ensuring the production quota is fulfilled might be beneficial in such a context.

## 8.5 Distribution of Revenue

Figure 48 indicates the distribution of revenue from production bids<sup>67</sup> in the forward market and producing hydrogen for the best performing learned model, the deterministic model and the hindsight benchmark. Note that the settlement in the balancing stage is not included in the figure, since it does not make sense to differentiate if the settlement cost or compensation was due to forward market participation or hydrogen production.

<sup>67</sup>Consumption bids (negative forward bids) are not considered in the figure, which is why the combined revenues in the figure are higher than the final scores of the models.

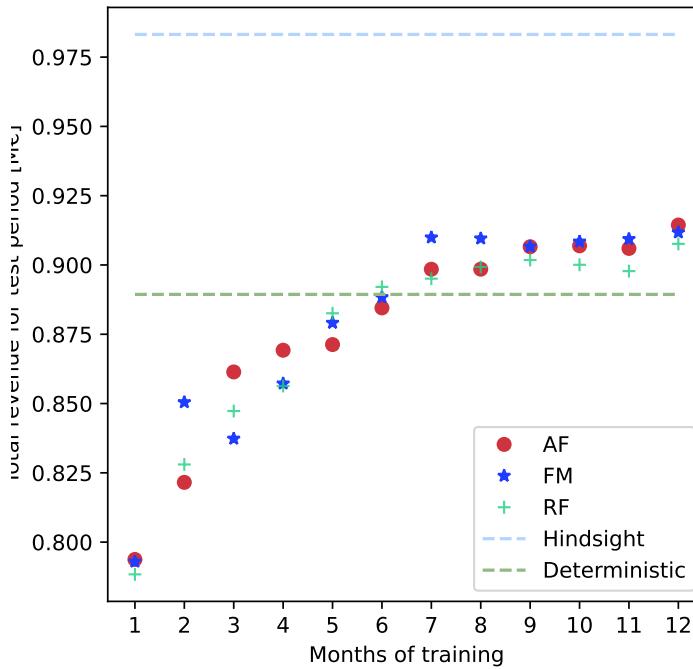


**Figure 48:** Distribution of revenue between hydrogen and forward market

It is seen that the vast majority of the revenue is gained from hydrogen production. This can be explained to a large degree by the low forward prices in the testing period, that had a mean value of 28.43 €/MW (34.18 €/MW for the entire period), which is a fair bit lower than the hydrogen price at 35.20 €/MWh. This relation means that more often than not, producing hydrogen will result in a larger revenue than selling the production in the forward market.

With hydrogen production making up such a large fraction of the total revenue gain, the models are effectively prioritizing to learn when to produce hydrogen, and might not actually perform particularly well in the forward market at all when considered isolated.

One way to shift the test to better show how the models perform in the forward market is to reduce the size of the electrolyzer, such that most of the production will have to be traded in the forward market instead of directed towards hydrogen production. This will also allow for an evaluation of the impact of the relative electrolyzer size. Figure 49 shows the results for model **HAPD** along with the deterministic and hindsight model, but where the size of the electrolyzer is reduced from 10MW to just 1MW, while keeping the wind farm at 10 MW.



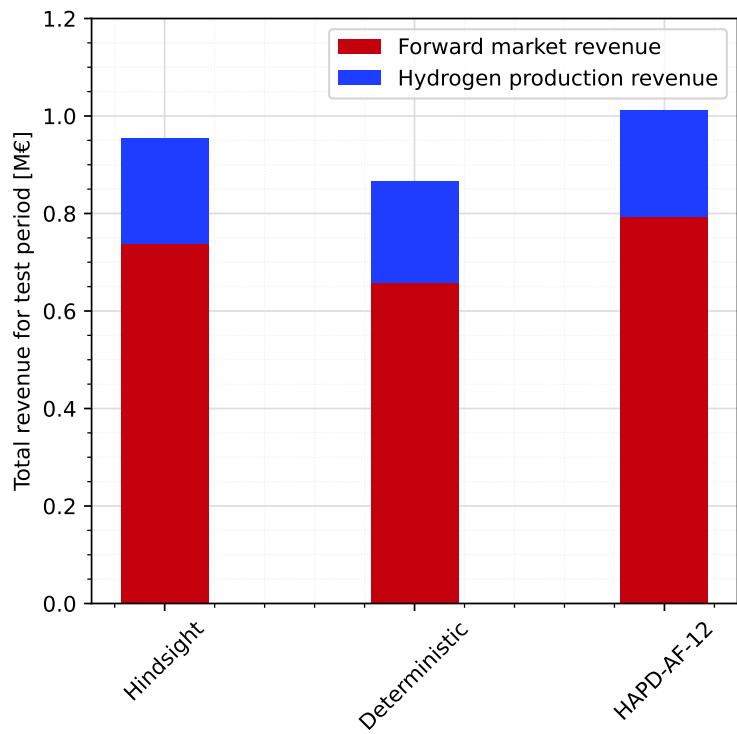
**Figure 49:** Total revenue with reduced electrolyzer size - **HAPD**

The first thing to notice is that the learned model still outperforms the deterministic, which shows that the learned model is capable of outperforming the deterministic in both contexts<sup>68</sup>. A noticeable result is that it takes longer training periods for the learned model to outperform the deterministic. This could indicate that learning to perform well in the forward market requires longer training periods than learning to generate revenue from hydrogen production.

It is also noted that the **FM** feature vector generally performs better than the others for training periods of more than 7 months, which is consistent with both earlier findings in section 8.1 and the findings of the main inspiration paper [12], which is exclusively considering forward market bids.

Figure 50 shows the distribution of revenue for the best performing model **HAPD-AF-12** along with the hindsight and deterministic model.

<sup>68</sup>One being a hydrogen production oriented context, the other being forward market oriented.

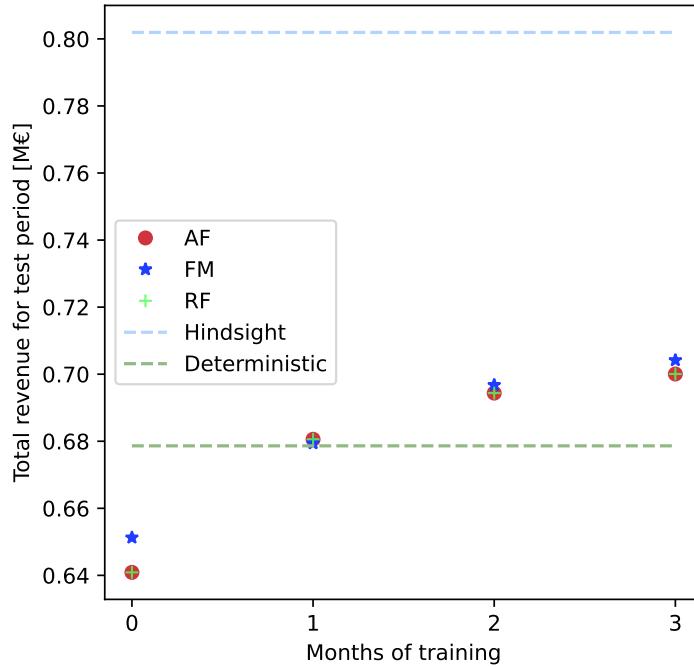


**Figure 50:** Distribution of revenue with reduced electrolyzer size

Figure 50 confirms that the forward market operation is responsible for the majority of the generated revenue when the size of the electrolyzer is reduced, and thus validates the conclusions drawn based on the results from figure 49.

## 8.6 Testing in 2022

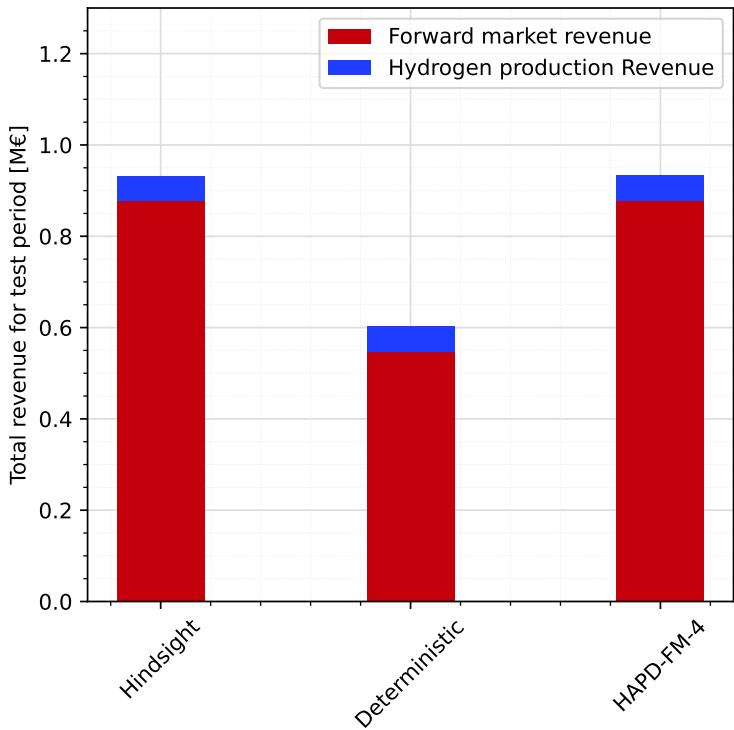
To perform a small test of generality of the developed models, the best performing model is tested on the 5 months of real-world data in 2022 provided by SG, that was used to create the synthetic forecasts. The first 4 months are used for training, and the final month is used for testing the models. The results using 2022 data is seen on figure 51.



**Figure 51:** Results for model **HAPD** on 2022 data

As seen on the figure, the learned model also outperforms the deterministic model on the 2022 data, and the trend showing better performance for longer training periods is the same as seen on figure 42. It is therefore expected that the learned model would perform even better given longer training periods.

An important difference between 2020 and 2022, is that the electricity prices were significantly higher in 2022, with a mean forward price of 361.92 €/MW in the final month used for testing (293.66 €/MW for the entire 5 months). This would be expected to cause the models to prioritize performing well in the forward market over hydrogen production, which is contrary to the 2020 results described in section 8.5. For the sake of curiosity, figure 52 shows the distribution in revenue from hydrogen and forward market bids, similar to figure 48, to see the impact of the different relation between forward price and hydrogen price. The best performing learned model on the 2022 data, **HAPD-FM-4**, is used for comparison.



**Figure 52:** Distribution of revenue for 2022 data

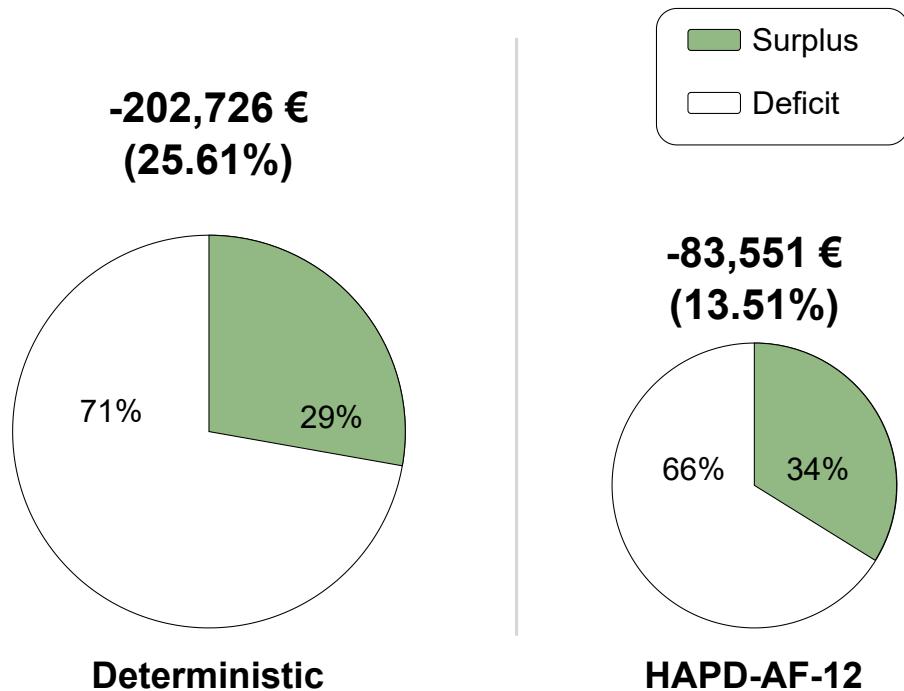
It is seen that as expected with the higher average forward price, the models now earn the majority of the revenue in the forward market, with the hindsight and **HAPD-FM-4** model earning  $\approx 6\%$  from hydrogen, and the deterministic earning about 10 %. The findings from section 8.5 that the forecasting model provides its value in a forward market oriented context is thus confirmed once again, with model **HAPD-FM** consistently outperforming models using the other feature vectors. Note that similarly to figure 48, figure 52 is not accounting for the balancing settlements and negative forward bids, so even though the learned and hindsight model earns the same from the forward market and hydrogen production (less than 2 % difference), the hindsight model has a 13.89 % higher final revenue than the learned model.

## 8.7 Balancing Settlements

Due to the dual pricing scheme, any deviations in the balancing stage will either be penalized or have a cost of zero. The balancing settlements can thus be a large source of loss for the models, and is worth an investigation. Figure 53 shows an illustration of different key results regarding balancing settlements for the deterministic model and the best performing feature-driven model **HAPD-AF-12**. The area of the circles represent the total amount settled<sup>69</sup>, and the green area represents how much of the settled amount was surplus production, the white area how much was deficit. Above each circle is indicated the net total of all settlements, where a (-) is prefixed to indicate that the settlements resulted in negative revenue for both models<sup>70</sup>. Underneath the amount is indicated how large a percentage loss in revenue the amount corresponds to.

<sup>69</sup>The sum of both surplus and deficit settlements in [€].

<sup>70</sup>Note that a loss would be incurred even if that net total was positive, since the compensation for surplus settlements has an opportunity cost compared to selling the production in the forward market.



**Figure 53:** Balancing settlements overview

Since balancing settlements will never result in a gain, the hindsight model has no balancing settlements and is therefore not included in the figure.

The smaller amount of settlements for the feature-driven model is expected, since it outperformed the deterministic model. An interesting result is that both models have  $\approx 2/3$  of the settlements being deficit. Since the feature-driven model is value-oriented, a bias in balancing prices<sup>71</sup> might be the cause of this "preference" towards deficit settlements. To compare the value of surplus and deficit settlements, a penalty  $\psi$  can be defined for each of them, as the difference between the forward price and the associated balancing price.

$$\psi_t^{UP} = \lambda_t^{UP} - \lambda_t^F \quad \forall t \in T \quad (154)$$

$$\psi_t^{DW} = \lambda_t^F - \lambda_t^{DW} \quad \forall t \in T \quad (155)$$

The penalty for deficit production is thus the premium  $\psi^{UP}$  that is paid in the settlement, and the penalty for surplus production  $\psi^{DW}$  is the opportunity cost associated with getting a lower compensation than would have been achieved in the forward market. Note that in each hour  $t$ , at least one of the penalties will equal zero.

Investigating the penalties for the entire dataset, it is found that 40 % of the time the grid is in a surplus status with  $\psi_t^{DW} > 0$ , and 24 % of the time the grid is in a deficit status with  $\psi_t^{UP} > 0$ . The remaining timesteps, the grid is balanced with both penalties equal to zero. To some degree this seems to explain why the feature-driven models incur more deficit than surplus settlements, since deficit settlements will be penalized less frequently than surplus settlements. However, the models are optimizing based on value, and if the expected value of the two types of settlements<sup>72</sup> is calculated, it is found that the expected value of incurring a deficit is higher than it is for surplus settlements<sup>73</sup>:

<sup>71</sup>Meaning one type of deviation being penalized more than the other.

<sup>72</sup>Which is identical to the mean of each penalty in this case.

<sup>73</sup>This relation is true when considering both the training data, the testing data, and the entire dataset combined. The values presented are for the entire dataset.

---


$$E[\psi_t^{UP}] = \frac{1}{|T|} \sum_{t \in T} \psi_t^{UP} = 6.17\text{€}/\text{MW} \quad (156)$$

$$E[\psi_t^{DW}] = \frac{1}{|T|} \sum_{t \in T} \psi_t^{DW} = 5.01\text{€}/\text{MW} \quad (157)$$

This leads to the expectation that *surplus* deviations would be preferred in general, since the expected value is lower. The results are thus contrary to the expectation in this regard, and the explanation needs to be found elsewhere.

When a deeper investigating was made into the timesteps with a deficit settlement for both the deterministic and the feature-driven model, a single pattern stood out: A deficit is incurred when the production forecast is higher than the realized value. Investigating the forecasts could provide insight to both the feature-driven models and the deterministic model regarding the tendency towards deficit settlements. It is no surprise that forecast errors result in deficit settlements, but the result is not that simple. The forecasts are distributed with a mean very close to zero, with the forecast being higher than the realized value in only 49.72 % of the cases<sup>74</sup>. When looking at the value of the forecasting error, it is also skewed such that forecasts are generally *lower* than the realized value. This means that both the amount and the frequency of the forecast errors point in the direction of realized surplus being the most common situation. The larger degree of deficit settlements therefore cannot be explained by forecasting errors, since the forecasting error statistics indicate that the majority of settlements should be surplus.

The realized production was generally lower in times of deficit settlements, with a mean value of 3.22 MW realized production, where the mean value for the entire test period was 4.51 MW of realized production. Very low amounts of realized production means that in order to fulfill the hydrogen production quota, power would have to be taken from the grid. However, since the forecast is not biased, this should not result in deficit settlements, but be accommodated by negative forward bids, meaning this relation does not provide any explanatory value either.

Other patterns were investigated, such as the distribution of negative vs. positive forward bids, the size of the forward bids, and the amount of hydrogen produced. All of these patterns showed the same distribution<sup>75</sup> for situations of deficit settlements as for the entire testing period, meaning no explanatory correlations could be found.

In spite of this extensive investigation, no clear direction for improvement was found based on the results for the balancing settlements. The conclusion is therefore simply that both types of models are apparently performing worse<sup>76</sup> when presented with lower forecasts, than with high forecasts, which is an important insight regardless of whether or not it can be explicitly explained by the data.

---

<sup>74</sup>The median thus being close to zero as well.

<sup>75</sup>Within a margin of significance of 10% for the statistics.

<sup>76</sup>Meaning they incur a larger loss in the balancing stage.

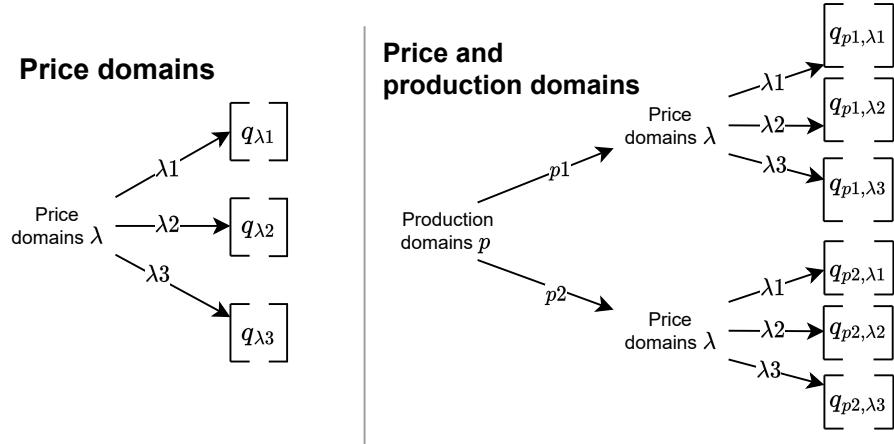
---

## 9 Discussion

The results of the studies in this project show that significantly larger revenues can be achieved by using a feature-driven model, compared to a deterministic model. The performance of the models was validated in different contexts, showing that the feature-driven models outperform the deterministic regarding both hydrogen production and forward market participation.

The *General* architecture performed better than the *Hourly* on short training periods, but the *Hourly* architecture outperformed the *General* architectures for training periods of more than 6 months, with the *Hourly* architecture reaching the highest total revenues. Even though the *Hourly* architecture reached the highest revenues, the *General* architecture might be more suited for data with a higher degree of non-stationarity, because of the better performance with shorter training periods. The *Hourly* architecture was implemented as a means to make the parameters conditional on the time of day, but this type of dependency could also have been achieved by making features specific on the time of day, for example by including the forecasts for other hours of the day in the feature vector instead of only using the forecast for the hour of bidding. This would allow the model to explicitly account for inter-temporal patterns in the features. But to retain the ability to also have different policies depending on the time of day, it would either require a large amount of dummy variables activating and deactivating the features depending on the time of day, or expanding the feature vector while using the *Hourly* architecture as well, both of which would entail a significant increase in the complexity of the model.

Introducing price domains improved both the stability and the total generated revenue significantly, and was a necessary measure for the feature-driven models to outperform the deterministic for both architectures. Adding more granularity to the price curve by increasing the number of price domains could be a direction of future work, but introducing domain specific areas for other features instead is expected to be more effective, because increasing the amount of price domains will most likely have diminishing returns. If the price days are distributed fairly smoothly, it will have diminishing returns because the statistics of the different domains will be more similar as the number of domains increase. To not have diminishing returns would require that a specific narrow domain had a large potential of revenue from a specific strategy (e.g. parameter values) that is significantly different from the surrounding domains. One way to find such a coincidence would be through a  $k$ -medoid clustering, as performed in subsection 3.1.2. When analyzing the forward price medoids, it was seen that there was not any medoid that showed significantly different characteristics than the others, and it is deemed unlikely for future data to have. The natural next step would therefore be to implement domain specific parameters based on the production forecast. Adding domain specific parameters to other features increases the complexity exponentially, since for example each production domain would have all the different price domains represented as subdomains as illustrated on figure 54. The researcher should thus exhibit great care and thorough testing when adding other domain-specific parameters.



**Figure 54:** Schematic of domain specific parameters multiplying

Retraining had no significant effect on any of the models. If the data the models were applied on had a higher degree of non-stationarity, retraining would be expected to have a larger positive impact on the performance.

An important limitation in the results of this report is the assumptions regarding the electrolyzer, presented in chapter 4. For future work, the cost and practical feasibility of keeping the status of the electrolyzer always being ON or in Standby could be evaluated, or the status modelling could be included in the model formulation. The consequence of assuming the electrolyzer was never OFF, was that real time adjustment could be performed on all the models. Implementing upwards adjustment from algorithm 14 had a small effect on the feature-driven model, but a significant effect on the deterministic. Upwards and downwards adjustment from algorithm 15 had a much greater impact, with the feature-driven model achieving 94.40% of the maximum possible gain from adjustment. Again the deterministic model benefited more than the feature-driven, however the feature-driven model still outperformed the deterministic after adjustment was applied. The incredibly large impact from applying adjustment was a significant result, since it effectively mediated a lot of the difference in performance of the different models. A significant assumption in the adjustment algorithms was that the balancing prices were known at the time of adjustment. This assumption was based on the fact that the system status could be predicted with a high probability, and the intraday market could be leveraged as well to forecast the balancing prices. However, no assessment was made on the risk associated with this assumption, which would be required to evaluate the true value of the adjustment algorithms. The results on real time adjustment thus show significant potential for large gains in revenue and improvement of poor performing models, but are simultaneously subject to oversimplified assumptions, both of which are reasons for further investigation of the mechanism in future work.

The balancing settlements of the models can be a tool to understand how the models can be improved. Both the feature-driven and deterministic models showed a tendency to primarily incur deficit settlements, which could not be explained by the performed investigation. Expanding this investigation would be a natural direction for future work, since it directly addresses the source of sub-optimality in the models.

Including other features could be a possible method of improving the performance further. Other than features that might improve the forecasted production, it might be relevant to include features containing information about the future system status, in order to leverage information about possible future balancing prices in the forward bid and hydrogen schedule.

Another possibility would be to include non-linear combinations of features, which would allow the models

---

to capture non-linear patterns in the data while keeping the parameters linear. Linear parameters allows for lesser computational effort, as well as simpler and more stable mathematical models, and is therefore to be preferred compared to non-linear parameters.

Throughout the report, a dual pricing scheme was used for the balancing settlements. A single pricing scheme was implemented in Denmark in 2021, so the models would operate in a different context if they were applied today. The shift to a single pricing scheme can to some degree be thought of as a simplification, since it reduces the balancing prices to a single value with no constraints. This means that the models can be expected to perform fairly well if implemented as they are, because the only difference is that a net gain is incurred when real-time deviation is in the grid's favor, instead of a zero gain as is the case in a dual balancing scheme. The results demonstrated in section 8.6 on 2022 data thus corresponds to a lower bound on how the model would have performed if it was implemented in a real-world context with a single pricing scheme. The main difference worth noting is that the hindsight model from the results in section 8.6 does *not* represent an upper bound on the total revenue in a single pricing scheme, since the possibility of arbitraging between the forward market and the balancing stage leads to new potential gains.

---

## 10 Conclusion

This project investigated how feature-driven models can be used to operate an electrolyzer unit in combination with a wind farm. The project used the methods from [12] as a starting point for the operation of a wind farm, and added the possibility of hydrogen production in the mathematical formulation of the problem. The developed models produce both the bids in the forward market and the hydrogen production schedule as a combined output.

The feature-driven models were tested with a variety of features consisting of different production forecasts, and produced a price curve to achieve a dependency on the forward price in the realized forward bid and hydrogen production.

The feature-driven models were evaluated against a deterministic approach, as well as several benchmarks used to indicate upper bounds on performance. The maximum possible revenue for considered period is given by the hindsight benchmark in algorithm 16. The deterministic model in algorithm 2 achieved 90.22% of the maximum possible revenue, while the best performing feature-driven model **HAPD-AF-12** in algorithm 11 achieved 96.27%.

Implementing real-time adjustment showed potential for a significant increase in revenue, with the "Upwards and downwards adjustment" algorithm 15 showing the best results. When applied to the deterministic model, 97.19 % of the maximum possible revenue is achieved, and when applied to the best feature-driven model **HAPD-AF-12**, 97.79 % of the maximum possible revenue is achieved. The difference in performance between the models is thus reduced significantly by applying adjustment.

Although the results presented are valid specifically for wind farms and hydrogen production, the developed mathematical formulation and associated models are not necessarily limited to operating wind farms and electrolyzers, but will generalize to any intermittent source of power production combined with a consumption unit. This combined operation is expected to become more relevant in future years, as both the amount of solar parks and wind farms increases, in combination with the expansion of ptX technologies.

Integrating ptX technologies with intermittent production units is both a means to a more sustainable energy sector, as well as a source of increased revenue for market agents. By combining hydrogen production with power production, the producer can effectively cap their minimum earnings from power production at the price of hydrogen, by directing power to the electrolyzer in times of low electricity prices. This serves both as a means of stabilizing the grid, since low price points are a sign of a large amount of available production, and as a means of reducing the risk on investments in the technology, which will hopefully motivate investors to participate in a green transition of the energy sector.

$$d(x_1) = \begin{cases} 1 & \text{if } x_1 < x_{d1} \\ 2 & \text{if } x_{d1} \leq x_1 \leq x_{d2} \\ 3 & \text{if } x_1 > x_{d2} \end{cases} \quad (158)$$

$$h(t) = \mod(t, 24) \quad (159)$$

$$\mathbf{q}^\top \mathbf{x}_t = \sum_{n=1}^N q_{h(t)}^n x_t^n \quad (160)$$

---

$$\mathbf{x}_t = \begin{bmatrix} \vdots \\ x_d \\ \vdots \end{bmatrix} \quad (161)$$

---

## A Appendix

### A.1 Computer specifications

The machine used for all computations in this project has the following specifications:

- Model: ASUS ZenBook 14 UX325 64 bit
- Processor: AMD Ryzen 7 5700U with Radeon Graphics - 1801 Mhz, 8 Cores, 16 Logical Processors
- RAM: 16 GB

### A.2 Calculation of Wasserstein distance

Table 3 shows the sum of the Wasserstein distance between all pairs of medoids for each year:

$$d_{\text{fromYear}, \text{toYear}}^{WS, total} = \sum_{i=1}^k \sum_{j=1}^k d^{WS}(\text{medoids}_i^{\text{fromYear}}, \text{medoids}_j^{\text{toYear}}) \quad (162)$$

Where  $d^{WS}(\cdot)$  is the Wasserstein distance function from the SciPy.stats library in Python, and fromYear & toYear correspond to the rows and columns in table 3 respectively. The metric being calculated  $d^{WS, total}$  is thus the distance between the medoids of the different years, and not the entire distributions of the price days. This means the distance between the medoids of a single year can also be computed by equation (162), which is not directly comparable against the distance between years, but can be compared to the "self-distance" of other years as a relative measure of how spread out the medoids are, and thereby also the distribution. This "self-distance" is the used metric to quantify the spread in figure 7. Note that none of the values have a particular meaning in and of themselves, but should be used only as a relative measure of similarity, where some pairs of years can be said to be more similar than others.

fromYear	toYear = 2019	toYear = 2020	toYear = 2021
2019	135	378	1146
2020	378	354	1432
2021	1146	1432	1090

**Table 2:** Wasserstein distances between medoids

fromYear	toYear = 2019	toYear = 2020	toYear = 2021
2019	11	26	90
2020	26	27	104
2021	90	104	86

**Table 3:** Wasserstein distances between scaled medoids

The distances used in figure 7 are normalized such that the distance between 2020 and 2021 is 100.

### A.3 Hydrogen purchase agreement

This section shows an excerpt from a Hydrogen Purchase Agreement (HPA) between Coffeyville Resources Refining & Marketing, and Coffeyville Resources Nitrogen Fertilizers. The HPA is found in [14].

**EXHIBIT B****ANALYSIS, SPECIFICATIONS AND PRICING FOR HYDROGEN**

<b>Hydrogen</b>	
- Gaseous	
- Purity	not less than 99.9 mol.%
- Flow	
- Pressure	450 psig $\pm$ 30 psi
- Carbon Monoxide	less than 10 ppm
- Carbon Dioxide	less than 10 ppm
- Committed Volume	90,000 mscf of Hydrogen per Month with the intent of providing 3,000 mscfd, ratably, to Fertilizer Company
- Excess Volume	Up to 60,000 mscf of Hydrogen per Month, or more upon mutual agreement of the parties, with the intent of providing up to an additional 2,000 mscfd to the Fertilizer Company.
- Monthly Fee	Fertilizer Company will pay Refinery Company a Monthly Fee equal to the sum of the: <ul style="list-style-type: none"> <li>• Monthly Fixed Fee, plus</li> <li>• Monthly Variable Fee, plus</li> <li>• Monthly Excess Fee.</li> </ul>
- Monthly Fixed Fee	The initial Monthly Fixed Fee is \$185,400. The Monthly Fixed Fee is equal to all Fixed Costs and Capital Costs associated with producing 90,000 mscf of Hydrogen per Month (see the formula below). <p style="text-align: center;"><b>Monthly Fixed Fee = (FC + CC) * 90,000 mscf per Month</b></p> <p><b>Monthly Fixed Fee (\$185,400.00) = Fixed and Capital Costs for Committed Volume (initially \$2.060/mscf Hydrogen) * Committed Volume (90,000 mscf of Hydrogen per Month)</b></p> <p>The Parties agree that after the Initial Term and during any Renewal Term, the Monthly Fixed Fee will be reduced to \$56,250.00 which equals the Fixed Costs associated with producing 90,000 mscf of Hydrogen per Month (see the formula below).</p> <p style="text-align: center;"><b>Monthly Fixed Fee = FC * 90,000 mscf per Month</b></p> <p><b>Monthly Fixed Fee (\$56,250.00) = Fixed Costs for Committed Volume (initially \$0.625/mscf Hydrogen) * Committed Volume (903,000 mscf per Month)</b></p>

- Fixed Costs or FC	Initially \$0.625 / mscf of Hydrogen or the fixed costs of producing one mscf of Hydrogen
- Capital Costs or CC	\$1.435 / mscf of Hydrogen or the capital costs of producing one mscf of Hydrogen
- Monthly Variable Fee	<p>The Monthly Variable Fee is equal to the total monthly mscf of Hydrogen received by Fertilizer Company (up to the Committed Volume) ("RCV") multiplied by the sum of 52% of the Natural Gas Price plus Other Variable Costs per mscf (see the formula below). [Note: 52% is used based upon the estimate of 11,180 mscfd of natural gas needed (for feed and furnace) to produce 21,500 mscfd of Hydrogen.]</p> $\text{Monthly Variable Fee} = \text{RCV} * [(\text{NGP} * .52) + \text{OVC}]$ <p>Therefore, if Fertilizer Company received the entire Committed Volume and the Natural Gas Price was \$3.00, the Monthly Variable Fee would be \$100,800.</p> $\text{Monthly Variable Fee (\$100,800)} = \text{RCV (90,000 mscf per Month)} * [(\text{NGP} * .52) + \text{OVC}] / (\$3.00 * .52) - \$0.44$
- Monthly Excess Fee	<p>The Monthly Excess Fee is equal to the total monthly Excess Volume received by Fertilizer Company ("REV") multiplied by the sum of 52% of the Natural Gas Price plus Other Variable Costs and Fixed Costs per mscf (see the formula below).</p> $\text{Monthly Excess Fee} = \text{REV} * [(\text{NGP} * .52) + \text{OVC} + \text{FC}]$ <p>Therefore, if Fertilizer Company received the maximum Excess Volume and the Natural Gas Price was \$3.00, the Monthly Excess Fee would be \$104,700.</p> $\text{Monthly Excess Fee (\$104,700)} = \text{REV (60,000 mscf per Month)} * [(\text{NGP} * .52) + \text{OVC} + \text{FC}] / (\$3.00 * .52) - \$0.44 + \$0.625$

- Monthly Adjusted Fixed Fee	<p>The Monthly Adjusted Fixed Fee is equal to the Monthly Fixed Fee multiplied by a fraction, the numerator of which is the RCV for the applicable Month and the denominator of which is the Committed Volume.</p> <p><b>Monthly Adjusted Fixed Fee = Monthly Fixed Fee * (RCV/Committed Volume)</b></p> <p>Example 1: If the CCR is down in the Refinery for 20 days and for the remaining 10 days of the applicable Month, Fertilizer Company receives a total of 50,000 mscf of Hydrogen for the applicable Month, Fertilizer Company pay a Monthly Adjusted Fix Fee of \$103,000.</p> <p><b>Monthly Adjusted Fixed Fee = 185,400 * (50,000 mscf/90,000 mscf) = \$103,000</b></p> <p>Example 2: If the CCR is down in the Refinery for 10 days in the applicable Month and for the remaining 20 days of the applicable Month, Fertilizer Company receives a total of 100,000 mscf of Hydrogen for the applicable Month, then Fertilizer Company will not receive a pro-rata reduction and be required to pay the full Monthly Fixed Fee and Monthly Variable Fee and Monthly Excess Fee.</p>
- Natural Gas Price or NGP	Natural gas measured at a per mmbtu rate based on the price for natural gas actually paid by Refinery Company for the month preceding the sale.
- Other Variable Costs or OVC	<p>-\$0.44</p> <p>The sum of the steam benefit (-\$0.54) plus power costs (\$0.03) plus chemical costs (\$0.07).</p>
- Escalation	<p>The Fixed Costs set forth in this Exhibit B are subject to change annually commencing January 1, 2018 and each anniversary thereafter. The Fixed Costs will be adjusted using the Bureau of Labor Statistics ("BLS") Employment Costs Index Average for Private Industry Workers (all workers) published in December of the previous year.</p> <p>For example, if the Fixed Costs for 2017 is \$.0625 and the BLS index published for December 2016 is 2.0% (not a real value), the 2018 Fixed Costs would be calculated as follows:</p> <p><b>2018 FC = 2017 FC + 2017 FC(2016 BLS Index published in December 2016)</b></p> <p><b>2018 FC = .0625 + .0625(2%) = .06375</b></p>

#### A.4 Proof of non-negative adjustment

It is not necessary to apply the  $(\cdot)^+$  operator to equation (152) to ensure that  $p_t^{adj} \geq 0$ , as proved by the following relations. Equation (152) implies equation (163) directly. The definition of  $p_t^{wanted}$  in equation (150) implies the relation in equation (164). Equation (165) combines equations (163) and (164) to show that these relations combined would imply that  $p_t^{surplus} > p_t^{wanted}$ , in which case equation (152) is not activated, but equation (151) instead, and thus the case of  $p_t^{adj} < 0$  is not possible.

$$p_t^{adj} < 0 \Rightarrow p_t^{surplus} > p_t^H \quad (163)$$

$$p_t^{H*} \geq 0 \Rightarrow p_t^{wanted} \leq p_t^H \quad (164)$$

↓

$$p_t^{surplus} > p_t^H \Rightarrow p_t^{surplus} > p_t^{wanted} \quad (165)$$

#### A.5 Postponing Adjustment Algorithm

Another level of complexity in the adjustment algorithms introduces the possibility of postponing a scheduled production to a later hour as an augmentation to algorithm 15. The situation where algorithm 15 still contains potential for improvement, is the case where downward adjustment is preferred, but there is not sufficient surplus to adjust as much as is preferred:

$$p_t^{H*} \leq p_t^H \quad (166)$$

$$p_t^{surplus} < p_t^{wanted} \quad (167)$$

Instead of just using whatever surplus might be available (which could be nothing), it could be more profitable to postpone the production to a later hour. However, neither the realized production nor the balancing prices of any later hours are known, and therefore any postponing will incur a risk for the producer.

The act of postponing, what to base the decision on, and how to evaluate the associated risk, is a subject that alone could be investigated for several chapters. The method of postponing presented here will be as simple as possible, such that it might serve as a starting point for future work.

Forecasting the balancing prices becomes significantly less accurate each hour into the future, due to the intrinsic randomness of the system status<sup>77</sup>. Forecasting balancing prices up to 23 hours into the future<sup>78</sup> is therefore an infeasible task. When deciding whether or not to postpone, the first evaluation in the present algorithm is therefore the forward prices for future hours, which are known at the time of adjustment.

The forward prices for all future hours of the considered day are assigned to a set of considered forward prices  $\Lambda^F$ :

$$\Lambda^F = \{\lambda_i^F | \forall i \in [t..t^D + 23]\} \quad (168)$$

The smallest price in this set is assigned to the value  $\lambda_{min}^F$ , and its index to  $idx_{min}^F$ :

$$\lambda_{min}^F = \min(\Lambda^F) \quad (169)$$

$$idx_{min}^F = \text{getHour}(\lambda_{min}^F, \Lambda^F) \quad (170)$$

<sup>77</sup>See [15] for an analysis of this, specifically figure 2 showing the diminishing accuracy of a system status prediction more than a couple of hours ahead of time.

<sup>78</sup>As would be required to evaluate all possible options of postponing.

---

Where  $\text{getHour}(\cdot, \cdot)$  is a function that returns the relative time index of the first argument, in the set passed as the second argument, such that the time of  $\lambda_{\min}^F$  can be referenced by the index  $t + \text{idx}_{\min}$ . This operation is denoted as **step 1**.

The forward price of future hours are used as a proxy of the downward balancing price<sup>79</sup> in that hour. If the current downward balancing price is lower than the considered forward price,  $\lambda_t^{DW} < \lambda_{\min}^F$ , then it is expected that even though the decision rule prefers to adjust down, it will most likely not be cheaper to postpone the production, and this causes the result from algorithm 15 to be activated:

$$p_t^{adj} = p_t^H - p_t^{surplus} \quad \text{if } \lambda_t^{DW} < \lambda_{\min}^F \quad (171)$$

If this is not the case, it could potentially be cheaper to postpone the production, and so this is attempted. This price check concludes **step 2**.

Before the production can be postponed, it must be checked whether or not the electrolyzer has available capacity in the hour that is being postponed to. The free capacity available in the electrolyzer at time  $t + \text{idx}_{\min}$  is given as:

$$p_{t+\text{idx}_{\min}}^{free} = \bar{P}^H - p_{t+\text{idx}_{\min}}^H \quad (172)$$

If  $p_{t+\text{idx}_{\min}}^{free} \geq p_t^{wanted}$ , then all of  $p_t^{wanted}$  is postponed, the current optimal hydrogen production can be effectuated:

$$p_{t+\text{idx}_{\min}}^H \leftarrow p_{t+\text{idx}_{\min}}^H + p_t^{wanted} \quad (173)$$

$$p_t^{adj} = p_t^{H*} \quad (174)$$

Where the notation  $a \leftarrow a + b$  denotes that the value of  $a$  is updated to be the previous value of  $a$  plus the update  $b$ .

If  $p_{t+\text{idx}_{\min}}^{free} < p_t^{wanted}$ , then all of  $p_{t+\text{idx}_{\min}}^{free}$  is utilized by producing at maximum capacity, and  $p_t^{wanted}$  is updated to what is remaining after subtracting the postponed amount.

$$p_{t+\text{idx}_{\min}}^H = \bar{P}^H \quad (175)$$

$$p_t^{wanted} \leftarrow p_t^{wanted} - p_{t+\text{idx}_{\min}}^{free} \quad (176)$$

This concludes **step 3**.

Given that step 3 was completed, and  $p_t^{wanted} > 0$  after the update, the price of the hour that was postponed to is added to a set  $\Omega$ , which is a set of prices removed from the set  $\Lambda^F$ . The amount of hydrogen production scheduled for later hours  $p_t^{planned}$  and consequently the surplus  $p_t^{surplus}$  is updated according to the changed schedule:

$$\Omega = \Omega \cup \lambda_{\min}^F \quad (177)$$

$$p_t^{planned} = \sum_{i=t}^{t^D+23} p_i^H \quad (178)$$

$$p_t^{surplus} = p_t^{produced} + p_t^{planned} - p^{DMin} \quad (179)$$

---

<sup>79</sup>Which is an obvious place to improve the algorithm for future work.

---

Afterwards the process is repeated by looping from step 1. This continues until either all of  $p_t^{wanted}$  has been postponed, or the price check in step 2 causes the result in equation 171 from algorithm 15 to be activated, terminating the loop.

All the steps combined are seen in algorithm 18:

---

**Algorithm 18** Postponing production

---

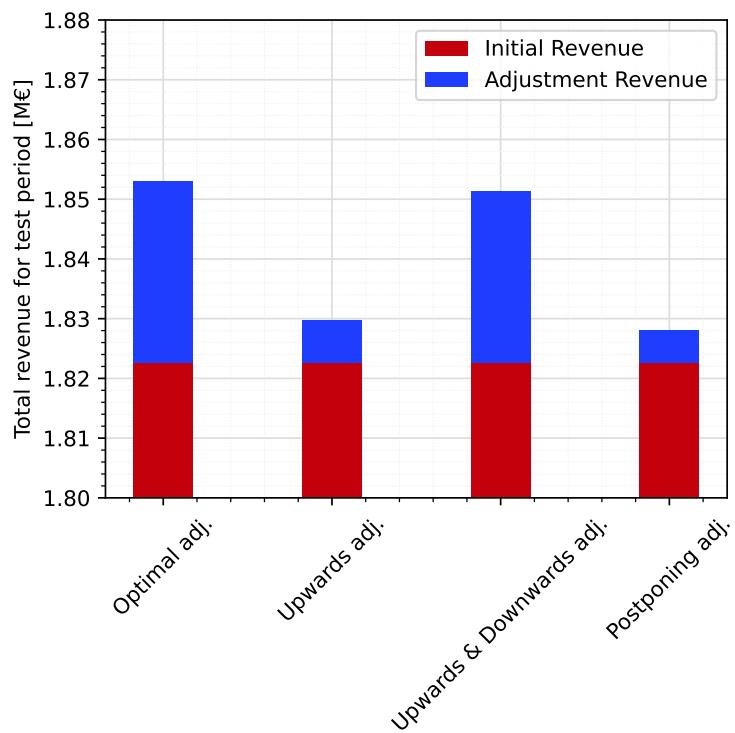
```

for each day  $t^D \in T^D$  do
    Initialize  $p_{t^D}^{produced} = 0$ 
    for each hour  $t \in [t^D, t^D + 23]$  do
        Receive model output:  $p_t^F, p_t^H$ 
        Receive balancing price forecasts:  $\lambda_t^{UP}, \lambda_t^{DW}$ 
        Realize production:  $E_t^{real}$ 
        Compute real-time deviation  $d_t = E_t^{real} - p_t^F$ 
        Compute optimal hydrogen production  $p_t^{H*} = \Pi(d_t, \lambda_t^{UP}, \lambda_t^{DW})$ 
        if  $p_t^{H*} > p_t^H$  then
             $p_t^{adj} = p_t^{H*}$ 
        else
             $p_t^{planned} = \sum_{i=t}^{t^D+23} p_i^H$ 
             $p_t^{surplus} = p_t^{produced} + p_t^{planned} - p^{DMin}$ 
             $p_t^{wanted} = p_t^H - p_t^{H*}$ 
            if  $p_t^{surplus} \geq p_t^{wanted}$  then
                 $p_t^{adj} = p_t^{H*}$ 
            else
                 $\Omega = \emptyset$ 
                while  $p_t^{wanted} > 0$  do
                    (Step 1)
                     $\Lambda^F = \{\lambda_i^F | \forall i \in [t, t^D + 23]\} \setminus \Omega$ 
                     $\lambda_{min}^F = \min(\Lambda^F)$ 
                     $idx_{min}^F = \text{getHour}(\lambda_{min}^F, \Lambda^F)$ 
                    (Step 2)
                    if  $\lambda_t^{DW} < \lambda_{min}^F$  then
                         $p_t^{adj} = p_t^H - p_t^{surplus}$ 
                    exit while
                    else
                        (Step 3)
                         $p_{t+idx_{min}}^{free} = \bar{P}^H - p_{t+idx_{min}}^H$ 
                        if  $p_{t+idx_{min}}^{free} \geq p_t^{wanted}$  then
                             $p_{t+idx_{min}}^H \leftarrow p_{t+idx_{min}}^H + p_t^{wanted}$ 
                             $p_t^{adj} = p_t^{H*}$ 
                        exit while
                    else
                         $p_{t+idx_{min}}^H = \bar{P}^H$ 
                         $p_t^{wanted} \leftarrow p_t^{wanted} - p_{t+idx_{min}}^{free}$ 
                         $p_t^{planned} = \sum_{i=t}^{t^D+23} p_i^H$ 
                         $p_t^{surplus} = p_t^{produced} + p_t^{planned} - p^{DMin}$ 
                         $\Omega = \Omega \cup \lambda_{min}^F$ 
                    end if
                end if
            end while
        end if
    end if

```

---

The results of applying algorithm 18 to the best performing learned model<sup>80</sup> is shown in figure 55 along with the results of the other adjustment algorithms 14 and 15.



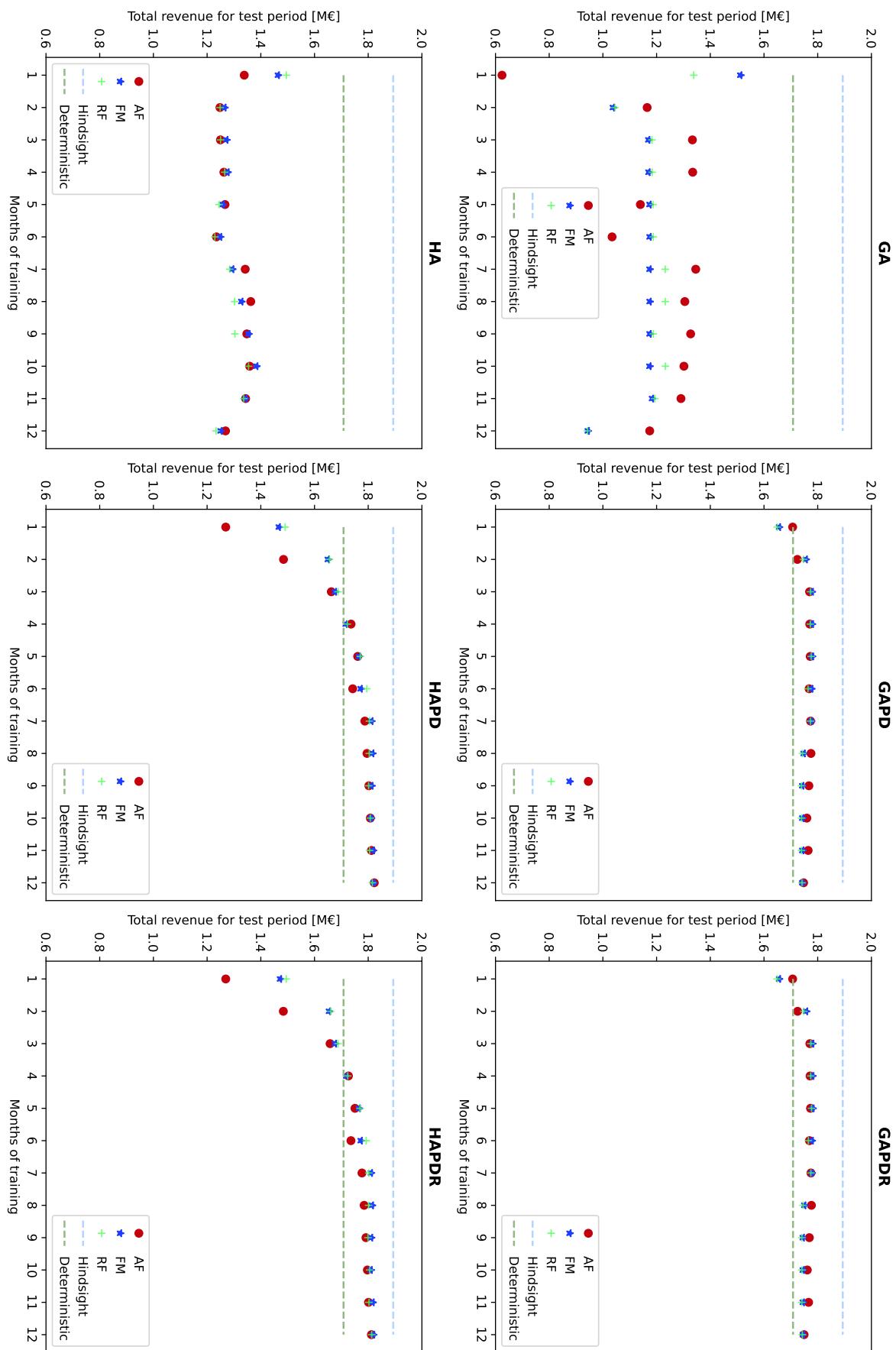
**Figure 55:** Results of postponing adjustment algorithm

#### A.6 Overview of results from section 8.1

An enlarged figure suitable for printing containing the results from figures 41 and 42 is seen in figure 56.

---

<sup>80</sup>Model **HAPD-AF-12**.



**Figure 56:** Enlarged overview of results  
Page x of xii

---

## References

- [1] Entso-e stated goals. <https://www.entsoe.eu/about/research-and-innovation/>.
- [2] Entso-e transparency platform. [https://transparency.entsoe.eu/?site\\_preference=normal](https://transparency.entsoe.eu/?site_preference=normal).
- [3] Af2021 - brintpriser. <https://energinet.dk/Analyse-og-Forskning/Analyseforudsætninger/Analyseforudsætninger-2021/>, Oct 2021.
- [4] The government's strategy for power-to-x, 2021. Danish Ministry of Climate, Energy and Utilities, [https://ens.dk/sites/ens.dk/files/ptx/strategy\\_ptx.pdf](https://ens.dk/sites/ens.dk/files/ptx/strategy_ptx.pdf).
- [5] Repowereu, an initiative under the european green deal. [https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/european-green-deal/repowereu-affordable-secure-and-sustainable-energy-europe\\_en](https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/european-green-deal/repowereu-affordable-secure-and-sustainable-energy-europe_en), Dec 2022.
- [6] Thomas Carriere and George Kariniotakis. An integrated approach for value-oriented energy forecasting and data-driven decision-making application to renewable energy trading. *IEEE Transactions on Smart Grid*, PP, 05 2019.
- [7] Hydrogen Denmark. Ve 2.0 - hydrogen and ptx-strategy (translated from danish). Technical report, Hydrogen Denmark - Brintbranchen, 2020.
- [8] Adam N. Elmachtoub and Paul Grigas. Smart "predict, then optimize". 2017. <https://arxiv.org/abs/1710.08005>.
- [9] Energistyrelsen. Technology data for renewable fuels. Technical report, Danish Energy Agency - Energistyrelsen, 2017 - latest version 2022.
- [10] Energistyrelsen. Technology data for energy storage. Technical report, Danish Energy Agency - Energistyrelsen, 2018 - latest version 2020.
- [11] Juan M. Morales, Miguel Á. Muñoz, and Salvador Pineda. Value-oriented forecasting of net demand for electricity market clearing. <https://arxiv.org/abs/2108.01003>, 2021.
- [12] Miguel Á. Muñoz, Juan M. Morales, and Salvador Pineda. Feature-driven improvement of renewable energy forecasting and trading. <https://arxiv.org/abs/1907.07580>, 2019.
- [13] Pierre Pinson. Distributionally robust trading strategies for renewable energy producers. <https://arxiv.org/abs/2209.08639>, 2022.
- [14] Mark A. Pytosh and Robert W. Haugen. Hydrogen purchase and sale agreement. <https://www.sec.gov/Archives/edgar/data/1425292/000142529217000039/uqnq12017exhibit101.htm>, Jan 2017.
- [15] Anders Skajaa, Kristian Edlund, and Juan M. Morales. Intraday trading of wind energy. *IEEE Transactions on Power Systems*, 30(6):3181–3189, 2015.
- [16] Erdogan Taskesen. distfit - Probability density fitting. <https://erdogant.github.io/distfit>, 2020. version 1.4.0.
- [17] Christopher Varela, Mahmoud Mostafa, and Edwin Zondervan. Modeling alkaline water electrolysis for power-to-x applications: A scheduling approach. *International Journal of Hydrogen Energy*, 46(14):9303–9313, 2021. <https://www.sciencedirect.com/science/article/pii/S036031992034725X>.

- 
- [18] Yi Zheng, Shi You, Henrik W. Bindner, and Marie Münster. Optimal day-ahead dispatch of an alkaline electrolyser system concerning thermal–electric properties and state-transitional dynamics. *Applied Energy*, 307:118091, 2022. <https://www.sciencedirect.com/science/article/pii/S0306261921013751>.