Science and
Technology
Facilities Council

Hartree Centre

National Grid ESO BSUoS project
Final report

Louise Butcher, Sarah Jackson,
Ravikanth Tadikonda

October 2023

Table of Contents

Executive Summary

- The Hartree Centre and National Grid ESO (NGESO) undertook a project to improve prediction of the Balancing Services Use of System Charge (BSUoS)
- The project focused on improving predictions up to 12 months ahead and moving to a daily forecast, using both NGESO's existing modelling approach as well as investigating new approaches.
- The project improved the understanding of the drivers behind NGESO's current model, and after thorough investigation of many different approaches identified some avenues for further investigation whilst ruling out many others.
- Training and validation datasets coincided with the wholesale price volatility starting in June 2021 which made it difficult to produce accurate forecast models.
- Within the period covered by the training dataset BSUoS costs were largely driven by Constraints, which is in turn driven by mainly by Renewable Proportion and Wholesale Cost.
- Building upon NGESO's existing model, multiple different approaches failed to produce a significantly improved forecast model, however one approach (Prophet model) showed a modest improvement.
- Work was also undertaken to improve the forecast model for Wholesale Price given its importance in driving BSUoS, a GARCH model was identified as showing promise.
- Future effort could be focused on:
    - Improving forecasting of Renewable Proportion which was identified as a more significant predictor than Wholesale Price (at least for Constraints costs).
    - Repurposing models from financial markets (e.g. GARCH) to directly predict costs (rather than using stochastic models to predict the wholesale prices).
- After investigating linear + ARIMA, Prophet, and Neural Network approaches, it was found that using daily data does not help in producing a better monthly forecast. However, using these approaches it was possible for the first time to produce forecasts using daily data, for daily costs, up to 30 days ahead. Recommendations have been made for future work in this area.
- Given the significant price volatility between the training and validation datasets, we recommend that NGESO review the performance of the Prophet and GARCH models as new data becomes available.

Introduction

This report describes the work undertaken by the Hartree Centre for National Grid ESO (NGESO) to improve prediction of balancing costs.

BSUoS is the Balancing Services Use of System charge, paid by transmission connected generation and demand to cover the cost of balancing the electricity system - for example running the national control room, frequency response arrangements, other ancillary services, and constraint costs (paying generators to turn on or off to maintain system security).

The tariff for BSUoS moved to a fixed-tariff system in April 2023, increasing the requirement for a good estimate of costs to enable the tariff to be set as accurately as possible.

In this project, Hartree worked with NGESO to improve the forecasting, especially in the short term (< 1 year), and to move towards daily forecasting.

The project was divided into individual work packages, which have already been documented in substantial technical detail. This report summarises and references those individual work package reports. In WP1, we performed exploratory data analysis to understand the data (NGESO BSUoS project – EDA.pdf and NGESO BSUoS project - data dictionary.pdf). In WP2 we built models of monthly data (WP2.pdf) and looked at wholesale price prediction (Wholesale Electricity Price Modelling.pdf), which in WP3 we looked at how to extend the modelling to daily data (WP3.pdf). In WP4 we used time series based neural network-based approach to look at daily data (WP4.pdf).

Assumptions

In discussion between Hartree and NGESO, it was agreed that:

- Hartree would focus on the short term forecasting
- Existing data was supplied pre-processed by NGESO, in the form in which is read into the main forecasting function in the existing NGESO code. Other data sources were looked at from the Open Data platforms where they can be downloaded, in particular data.nationalgrideso.com.

WP1 – Exploratory data analysis and data understanding

In this work package, we received the data from NGESO, worked together to understand what was contained in the data, and explored the existing NGESO code.

Existing code

The existing code supplied by NGESO was a single R file of 3122 lines. In order to make this easier to work with, we broke it down into multiple files, and focused on the function run_forecast_f(), which runs the short term forecast.

We then added a new main() function to run this, and later on, code to run various types of backcasting.

It would be beneficial in future to separate the code out further and to refactor it to have functions which perform a single task and group similar functions into packages. The code would also benefit from some unit tests to check that the individual functions produce results as expected.

Our changes over the course of the project are summarised in code_list.pdf

Data dictionary

We summarised the data in a data dictionary (NGESO BSUoS project - data dictionary.pdf) and looked at the potential of each data source to be used in daily as well as monthly forecasting. Where there was a data source which had the potential to be used for predictive modelling, but was not already being used, we explored with NGESO whether forecasts were available, or could readily be generated.

We found that the datasets mostly had daily data available already – the main problem would be creating daily forecasts for wholesale data.

We looked at the possibility of using inertia and system margin as inputs – it was determined that they were not straightforward to forecast, but could possibly be forecast if required.

Exploratory data analysis

We produced various Jupyter notebooks to explore and plot the different datasets. Some key findings are summarised here, but more detail is given in the work package report.

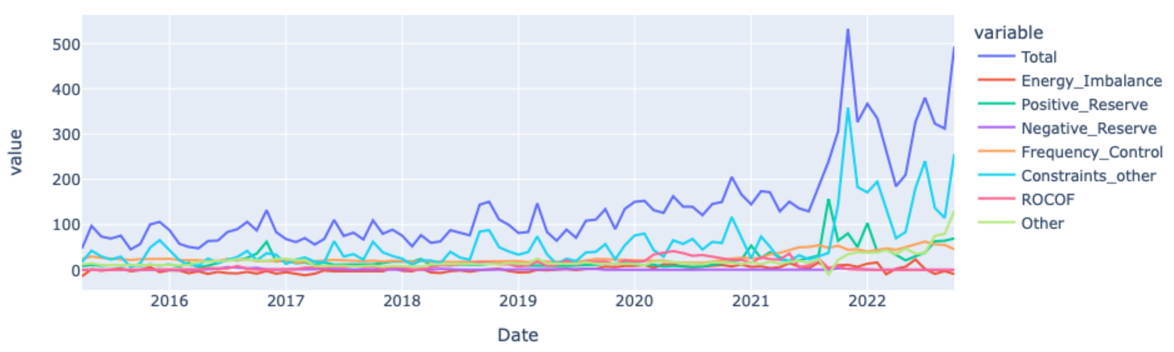


Figure 1: Monthly costs by type

Figure 1shows the monthly costs by category, and the total. From this we can see that Constraints_other is the largest contribution, and the total is largely driven by changes in Constraints_other, so this is a category of cost we should focus on. It also shows that the costs, and their volatility, increase significantly from around July 2021. Throughout the project, it has been found that this unprecedented behaviour makes forecasting very difficult.

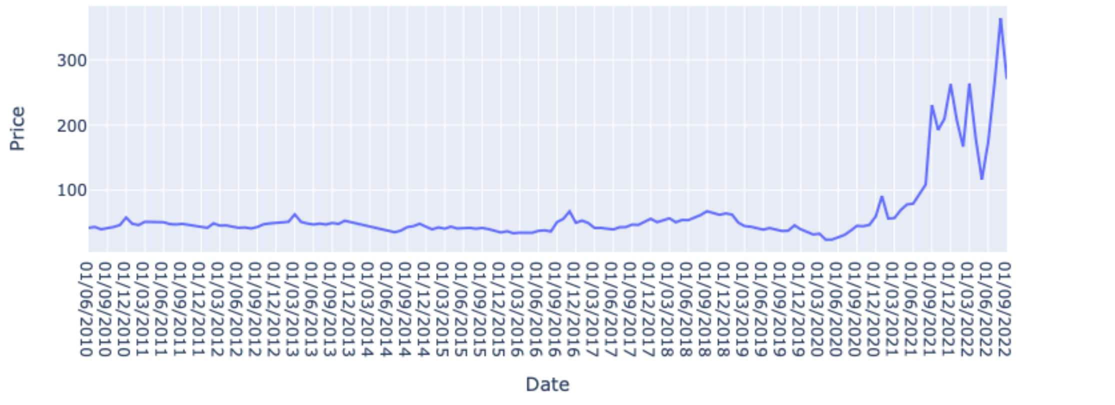


Figure 2: Monthly wholesale costs

Wholesale prices are one of the variables currently used to predict balancing costs. Figure 2 shows the plot of monthly prices with time. There is a sharp increase in values and volatility during the first half of 2021, as with the balancing costs, due largely to geopolitical factors (the war in Ukraine).

We then combined the different monthly datasets, mainly using the existing R code while saving intermediate results to file, other for daily and monthly data. This gave us the dataset of “actual” data which we used in WP2,3 and 4

Further detail and plots of other data sets can be found in the WP1 work package report, NGESO BSUoS project - EDA.pdf. Some other key observations from the data, and discussions about it were:

- In the data before 1/4/2017, the individual components do not exactly add up to the total – this was known to NGESO.
- Daily costs aggregate up to monthly costs as expected, once ROCOF and Constraints_other are added together for the monthly data
- Care is needed when using times in the n2ex data, as it uses both GMT and BST.
- Negative wholesale prices are valid data.
- The current wholesale price forecasts at 1-3 months out are highly dependent on the most recent actual prices and have significant fluctuations which do not correspond to what turns out to happen.

Additional datasets for inertia and system margin were explored, but there is no clear process to forecast these ahead of time.

Work was also undertaken in this work package to understand the existing models, to alter them to enable backcasting to use either forecast or actual values, and to setup a methodology for training and validating the data to get fair results. We setup walk forward validation to train on all data up to a point, and then forecast after that point, rolling forward the date of the split to try on different dates.

WP2 – modelling monthly data

Forecasting balancing costs

Predicting costs

We applied various models, outlined below, to forecasting balancing costs, focussing on the Constraints component. They have been optimised using backcasting for each individual component, and then applied in the original R framework with simulations.

We started with the model as used by NGESO, and found that the ARIMA aspect of it was not contributing to the modelling (illustrated in Figure 3 for the Constraints case, but analogous results were found for other components). The Linear Model contribution line (dark blue) is obscured by the Combined forecast line, except at the

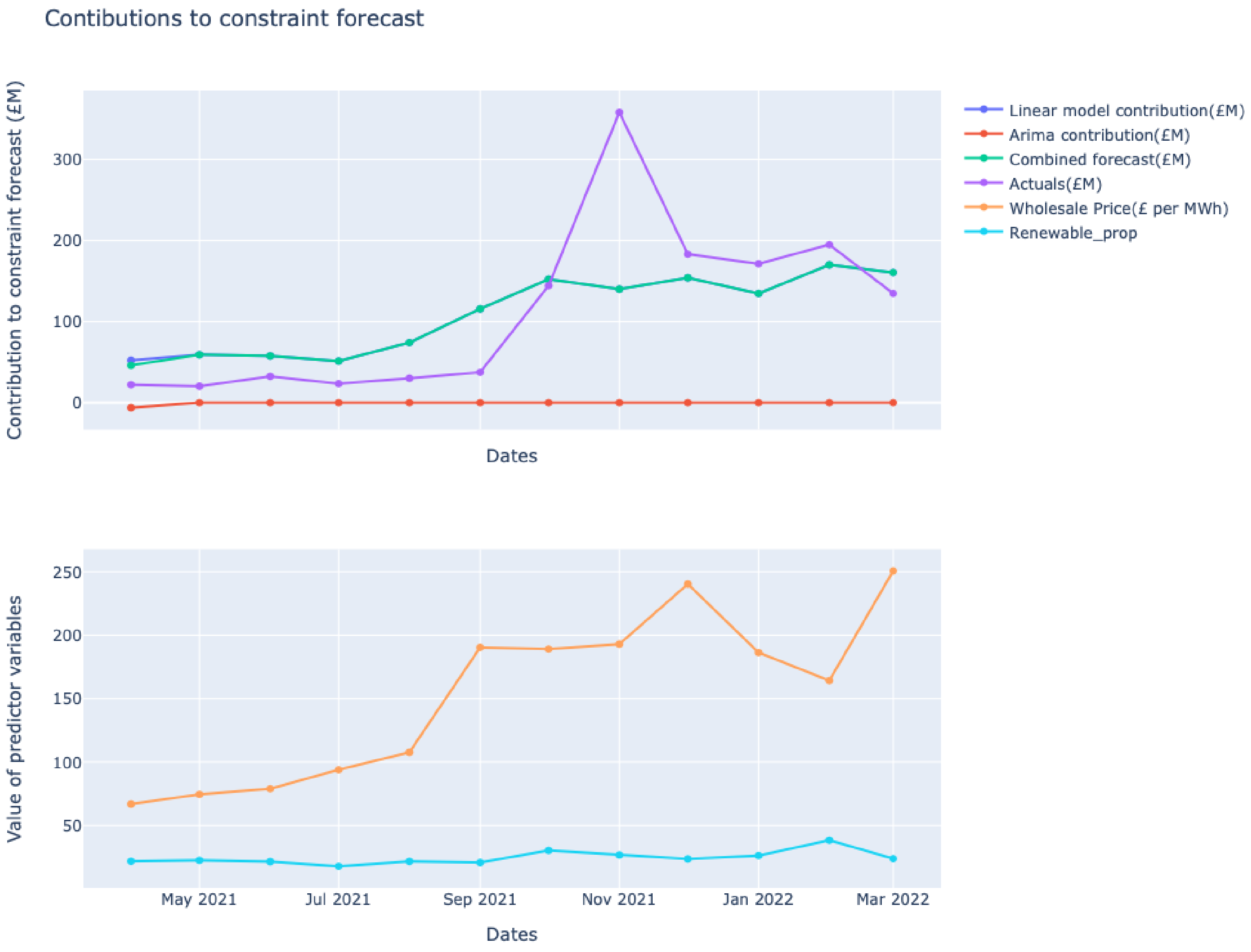


Figure 3: Contribution of linear model and ARIMA to constraints forecast

far left, as they have identical values except for the far left. The Combined forecast is the sum of the Linear model contribution and the Arima model contribution, but the Arima model contribution is zero for all but the first point.

The work in this work package focussed on trying to get a model which performs consistently better than this baseline when applied to past data.

Looking at the Constraints component, we explored various ARIMA models on the data, with and without regression and with and without making the data stationary first, but without much improvement. The components found by Auto ARIMA methods were inconsistent, and once the data was made stationary, there appeared to be no structure left to model.

We tried some standard machine learning models on the problem - Random Forest and Support Vector machines. These showed strong overfitting, and did not lead to improvement, as measured by Mean Absolute Error (MAE) on the validation set. We also experimented with an autoregressive term in the Random Forest model, but with no real benefit (again, using the largest Constraints component).

Additional variables

Using Constraints as the output variable, we explored using additional variables, both in linear models and particular in Random forest Random Forest has the benefit of providing “importances” as an output, which signify which are the most significant parameters. The new inertia variables were quickly rejected (not least because they are hard to forecast). Using other variables already in the data (Demand, PV, Embedded wind and BMU wind), as well as volatility calculation showed some promise, but not enough to give a better performing model than baseline. We looked at feature selection, which found that the renewable proportion is the most important variable, with the wholesale price next most important, but at lower importance.

Prophet

Given what we were seeing in deconstructing the data into seasonality and trend to make it stationary for ARIMA, a generalised additive model seemed promising, in particular Prophet. This method uses seasonality, trend and regressive components to forecast time series. We optimised the model using cross validation for each of the components. We still saw overfitting in this model, but the results were better than for the original linear model. The results relative to the original model are shown in Table 1.

Model	Component modelled	Model details	MAE (1/4/21-1/4/22 validation)	MAE (1/11/21-1/11/22 validation)	Mean MAE
Original Linear plus ARIMA	Constraints	-	48	81	65
Prophet	Constraints	Additive regressors,	40	60	50

		additive seasonality, 6 input variables			

Table 1: Performance of Prophet model compared to original model, using actual values for variables for prediction and using Constraints as output variable

Implementation in R

We implemented the Prophet model in R, using mostly the original code but with Prophet in place of ARIMAX, and keeping the simulations etc the same. The method of combining components into the total was also kept the same as in the original. We compared different versions of the model and selected a 2 variable model with Prophet's automatic hyperparameters as the best performing.

Using the unseen test set to compare the best Prophet model to the original linear model for the total costs, we see a small improvement in MAE from 78 to 73 based on using the actual values of all the variables as inputs. The full results are shown in Table 2. Given the modest improvement, further work is probably needed to test this to establish whether it is worth putting into production.

MAE for forecast vs actuals	Original		Prophet (2 variable)	
Cost category	Actual	Forecast	Actual	Forecast
Energy_Imbalance	16	16	16	16
Positive_Reserve	36	41	35	38
Negative_Reserve	1.3	1.2	1.3	1.2
Frequency_Control	18	19	19	20
Constraints	38	167	37	175
Other	34	35	35	35
Total	78	209	73	208

Table 2: Results on test data

Modelling Wholesale Electricity Prices

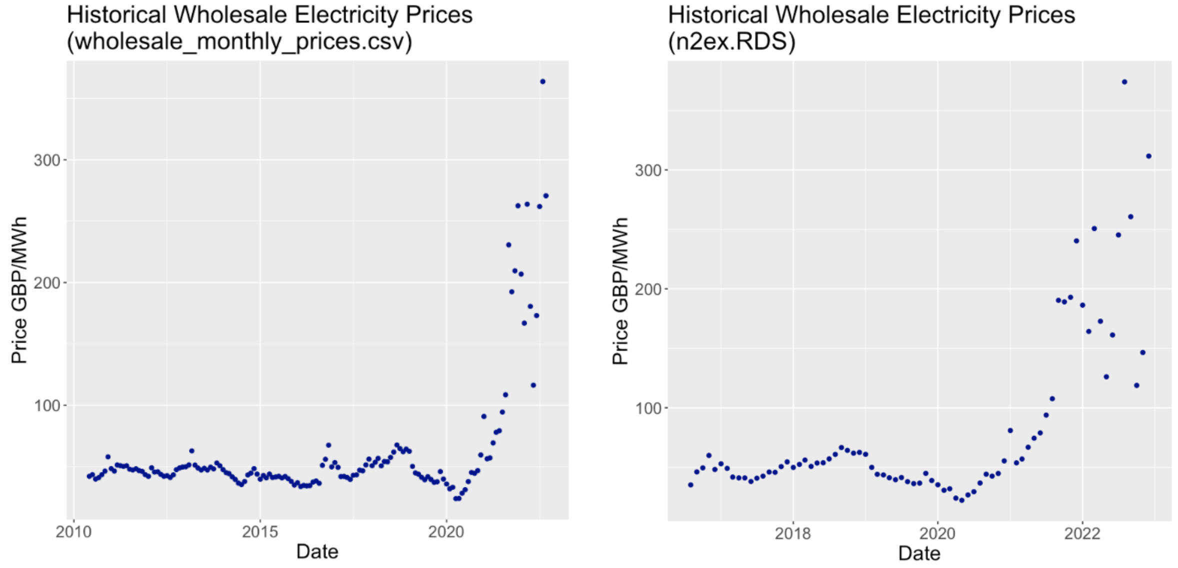
Introduction

Wholesale electricity price and renewable proportion of demand are used as predictor variables when forecasting balancing costs. To this end, the modelling of wholesale electricity price data was explored. The current modelling approach can be divided into two main stages. First, a central forecast is produced using an interpolation-based approach and forward wholesale electricity price data. In the second stage, a range of possible price trajectories are simulated using a stochastic differential equation model. Historic wholesale electricity price data is used to calibrate the parameters of this model.

The method of simulating possible price trajectories has been a focus of previous work. To this end, possible improvements to this model were not directly explored. Two alternative methods were investigated; first, alternative interpolation approaches for producing a central forecast using forward wholesale electricity price data, including how this impacts the subsequent price trajectory simulations which are centred around this forecast. The second explores the time series modelling of historic wholesale electricity price data to forecast future values.

Data Sources

The NGESO team use historic actual wholesale electricity price data in their balancing cost modelling and have recently moved to a new source of this data. The right plot of Figure 4 shows a plot of this new data mean aggregated from half-hourly resolution to monthly mean values. Data is available from August 2016 to present. Meanwhile, the left plot of Figure 4. shows the older source of wholesale electricity price data. Available at monthly resolution, it covers the period June 2010 to September 2022.


Historical Wholesale Electricity Prices (wholesale_monthly_prices.csv)

Historical Wholesale Electricity Prices (n2ex.RDS)

Figure 4: Historic wholesale electricity price from two available data sources

Moreover, forward wholesale electricity prices are provided to the NGESO team by an external company. A given set of values consists of forwards prices for the next few months, quarters, and seasons. Figure 5 shows an example of this data, obscured to avoid including the raw data in this report. Note this processing may result in the values and relationships between them being unexpected. It is assumed these values can be used as a reasonable prediction of wholesale electricity prices in the future.

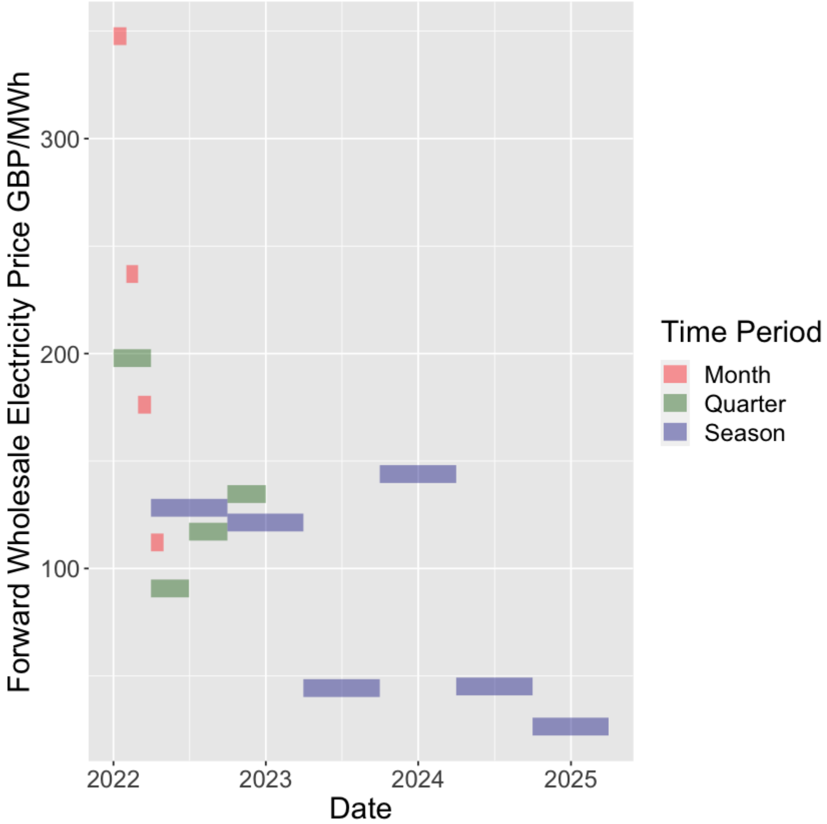


Figure 5: Example (obscured) unprocessed forward wholesale electricity prices (£/MWh)

Interpolation

Using forward wholesale electricity price data, several interpolation methods were explored including “flat” time-period, spline, and generalised additive model (GAM) based methods. However, choice of one method over another seemed arbitrary when compared to the actual values in recent years when prices have been more difficult to predict. However, some approaches do provide “smoother” forecasts which may be desirable, e.g., to aid communication of the uncertainty of predictions. Further detail is available in the wholesale electricity modelling report.

Time Series Modelling

The historic wholesale electricity price data seems to be correlated with itself (autocorrelation) and have non-constant variance (heteroskedasticity). ARFIMA-GARCH (autoregressive fractionally integrated moving average - generalised autoregressive conditional heteroskedasticity) models were explored as a possible option for modelling this data. This modelling approach allows for the incorporation of autoregressive and moving average terms when modelling the mean and variance. Further detail is available in the wholesale electricity modelling report. Once a model has been fitted, a large number of forecasts can be simulated. Figure 6 plots the quantile of a set of these of simulations. For this forecast date, the GARCH model seems to provide better coverage of possible price trajectories than the original method plotted in Figure 7.

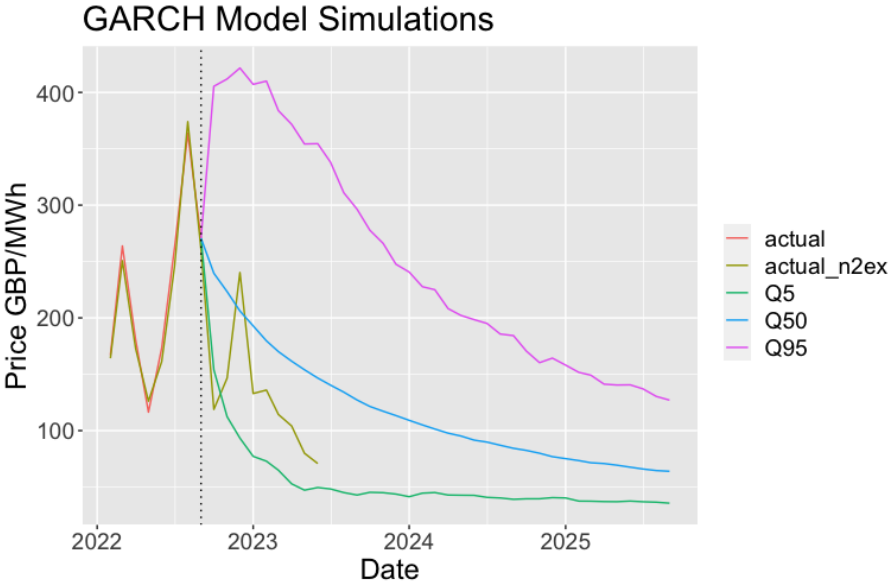


Figure 6: Quantiles over time of 1000 GARCH model simulations (36 point)

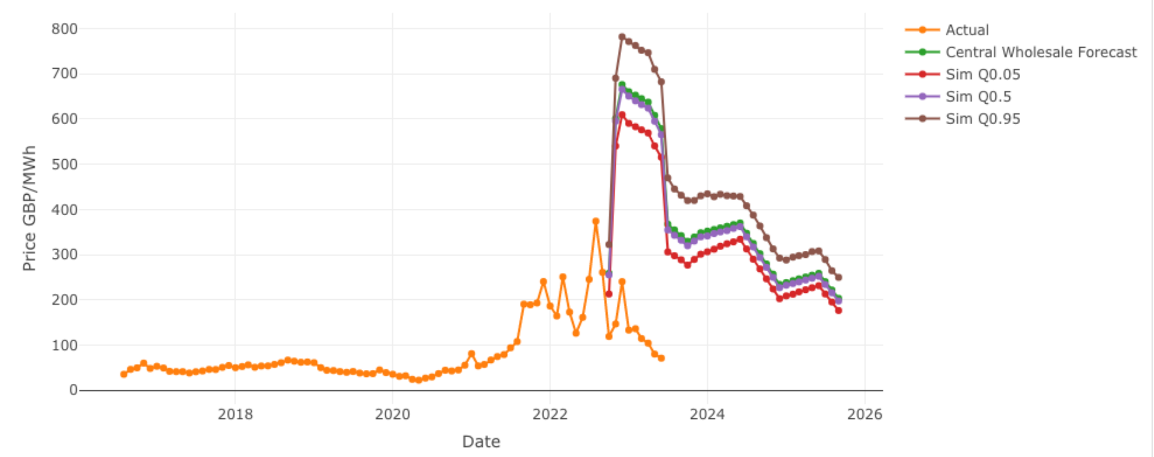


Figure 7: Original method forecast (2022-10-01 forecast date)

Discussion

The recent volatility in wholesale electricity prices is complex to model. Interpolation based approaches using forward prices rely on the assumption that this data is a reasonable and accurate prediction of future wholesale electricity price values. Meanwhile, time series approaches rely on adequately modelling underlying processes from the data. It is particularly difficult to model recent much higher prices. To what extent is this data driven by volatility or trend? Will these prices settle down, and if so, to what extent? How likely is this magnitude of volatility to occur in the future?

For the interpolation methods, it is recommended to consider whether having a "smoother" central forecast, also causing "smoother" simulations is desirable. The current method can give rise to relatively less smooth forecasts than some of the alternative interpolation methods. Greater smoothness may be more aligned with the uncertainty associated with future forecasts.

A GARCH modelling approach seems to have potential. It is recommended that this is explored further, especially as more data becomes available. By modelling variance using autoregressive and/or moving average terms, periods of volatility can be incorporated into modelling. However, the current data consists of one period of extremely relatively large volatility that is arguably still "in progress". It would be pertinent to ensure any model is generalisable and a good fit to the data overall, rather than just one period of volatility. The GARCH model could be used in parallel to the current model to compare output and assess performance.

WP2 Discussion and further work

Although we have obtained a modest improvement, it is not obvious that this is sufficient to merit changing the forecasting protocol at present. Further testing of the model is desirable to see how it performs as new data comes in, given the overfitting present when validating.

There is possibly scope for a little more optimising by trying different combinations of parameters, but the improvements will probably not be large.

If adopting these models, it will be worth revisiting the optimised hyperparameters periodically, but they need not be optimised every time the model is run.

The major problem in obtaining a better forecast is the changing nature of the cost data – during 2021 both wholesale prices and the resulting balancing costs grew as a result of geopolitical factors, and became more volatile. The timings are such that we ended up training a model on data before this period, and then validating it on data after this period. Machine learning models do not generalise well when training and validation data comes from different distributions. As more of this volatile data comes in, and prices start to settle, the behaviour to be forecast will have been seen already in the training data, and modelling will hopefully improve. We may already be starting to see this in the test data, which gives a better MAE than the validation set.

It seems that, certainly during this period, it may always be difficult to make forecasts. The balancing prices are driven in part by market sentiment and behavioural factors, with companies charging what the market will stand. The charges have a lot of unexplained variance due to seemingly arbitrary factors, and are therefore fundamentally hard to predict. In this respect they are much like share prices on financial markets.

A consideration in the future is that renewable proportion has come out as a more significant predictor than wholesale prices, at least for the Constraints costs that we focussed on. It could be useful to take another look at forecasting of the renewable proportion, perhaps using similar simulation techniques to wholesale forecasting, or incorporating weather forecasts.

A further area of future work could be to look at repurposing models from financial markets to directly predict costs (rather than using stochastic models to predict the wholesale prices). These may not make exact predictions, but may give an idea of the uncertainty or expected range in which the data might fall. GARCH is this type of model, but this is a large area to explore. GARCH had been explored for the wholesale price modelling and is useful to look at how the volatility is changing. There is an interesting comparison of different approaches on financial data in Orland et al.[1]

Having more data would certainly help in the model building – it would be interesting to revisit this analysis when time has passed and there is data on how the costs changed coming down from the peak. It would also be interesting to look at using AI methods such as LSTM on monthly data. WP4 has looked at using LSTMs on daily data, but there is not enough data to use them on monthly data.

Other future work could look at subsetting the data in different ways, to see under which circumstances the original and new models give good results (do we get good models if we only look at data before 2021?).

It might also be useful to feed in conclusions from the current NGESO 3MD project to see if that can be used to remove anomalous costs in the data and explore if it is easier to forecast what remains.

There may also be other datasets which are useful – oil prices might give a general idea of market conditions for instance. There may also be merit in looking at volumes and unit costs of balancing separately, as there may be different drivers.

WP3 – modelling daily data

In this work package, we looked at modelling daily data for two purposes – firstly an attempt to improve the forecasting of monthly data, and secondly to explore whether we can use daily forecasts to provide a finer-grained prediction.

The full details of the work can be found in the work package report WP3_21_8_23.pdf.

Using daily data to create better annual forecast

In this section, we experimented on a pre-joined dataset, using actual values of variables for prediction, and working on the Constraints component, as that is the

[1] Orlando G et al, Financial markets’ deterministic aspects modelled by a low-dimensional equation Sci Rep. 2022; 12: 1693. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8807815/

largest of the cost components. We calculated daily predictions over a year, and then totalled these up to make monthly predictions, for comparison to the work in WP2.

The first model to be tried was a linear plus ARIMA model, analogous to the monthly model (Figure 8).

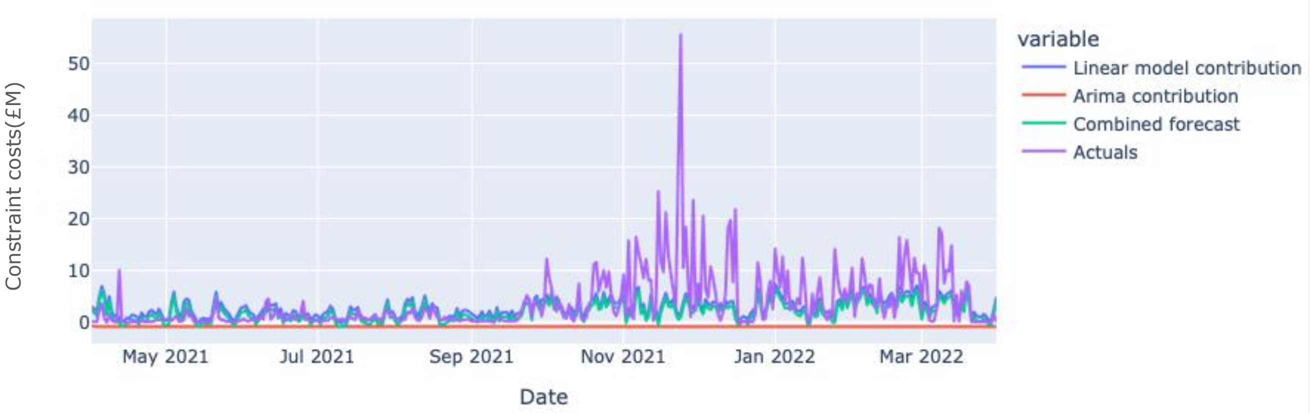


Figure 8: Linear model and ARIMA contributions to daily forecast

The ARIMA model did not add much to the forecast, as we found for the monthly data. Although this model captures the general shape of the data, and looks quite good over the earlier period, it does not capture the large spikes in late 2021.

We then tried building a Random Forest model for the regression component, as for the monthly case, but the R^2 values (the quality of fit of the Random Forest model) were poor on unseen data, and the overall MAE for the 12 month forecast was no better.

We then built generalised additive models using Prophet, as for the monthly data, optimising the hyperparameters.

None of these daily methods produced a model which was better than the Linear + ARIMA model, which was very close to the original WP2 performance. A summary of the results is shown in Table 3. More detail is given in the full workpackage report.

	MAE 12 months 1/4/21 date	MAE 12 months 1/11/21 date	Mean MAE 12 months
Original monthly linear + ARIMA (from WP2)	48	81	64.5
Best monthly model from WP2	40	60	50
Daily version of linear plus ARIMA, aggregated to monthly	62	69	65.5
Best daily model from WP2, aggregated to Monthly,	53	82	67.5

Prophet, 6 variable, default hyperparameters, additive regressors			

Table 3: Comparison of daily to monthly results from WP3 with results from WP2

Forecasting daily data up to 31 days

NGESO also wish to be able to calculate daily costs over a short time period for their own sake. After discussion, we picked a period of 31 days as the timeframe of interest over which to optimise the forecasts. It was also required that we be able to measure the performance of the models by the number of days ahead the forecast is made. We looked at Total costs, as this is expected to be more accurate than individual component, and used the actual values for input variables.

We built a Prophet model using all data available up to our cutoff point, and then predicted for the next 31 days using actuals as the input variables. We built a range of different models with different hyperparameters and found that using six input regressor variables consistently gave the best results. Figure 9 shows sample output, measuring the cumulative MAE over different numbers of days up to 31.

Test data - 31 day forecasting

We used the data after 1/11/22 to test the best model against a baseline linear model. We used the same walkforward approach as above, training the model on all data up to a cutoff date, and then forecasting for the next 31 days, using cutoff dates from 1/1/22 to 30/4/23 looking at the Total component and both cumulative and spot metrics. Results are shown in Table 4.

For the linear model, we used the same linear regression plus autoarima model as for the daily->monthly experiments, plus autoarima, using Price and Renewable_prop as the input variables. In both cases, the actual data is used for the input variables.

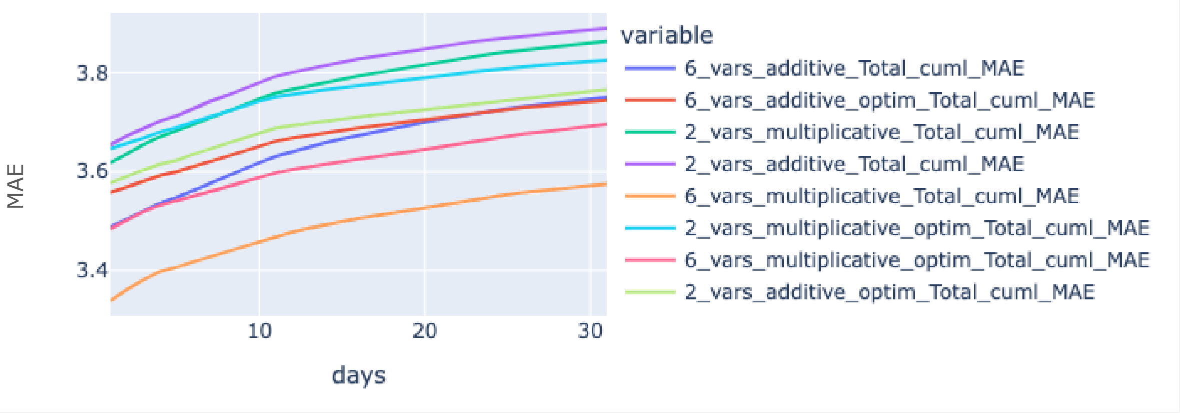


Figure 9: Cumulative MAE by number of days

	MAE (cumulative over 31 days)	MAE (spot – day 31)	R2 (cumulative over 31 days)	R2 (spot – day 31)	MAPE (cumulative over 31 days)	MAPE (spot – day 31)
Linear + arima	3.28	3.07	0.41	0.46	0.52	0.57
Prophet model 6 hyperparameters	3.23	3.43	0.43	0.31	0.51	0.58

Table 4: Comparison of Linear +Arima and Prophet models for forecasting over 31 days

Discussion and further work for WP3

Using daily data does not help in producing a better monthly forecast. It is not unusual in time series predictions of this type to find that it is easier to forecast on the more aggregated data, as this smooths out outliers and anomalies. In this case (as seen in WP1), the cost and wholesale price data is particularly “spikey” – a month with high wholesale prices or costs does not have exceptionally high costs for every day. It was thought at one point that maybe these outliers were driving the monthly peaks, and that this might be a good target for future work, but an analysis showed that this was not the case.

As with the monthly data, there may be some model we can borrow from financial markets to make a better forecast, but it is unlikely to be deterministic and may just give different types of simulations.

Future work could also look more closely at the residuals after the data has been fit – if the residuals have the statistical properties of white noise, then it is unlikely that a better model can be produced. Tests for noise might be:

- The residuals have a distribution around a zero mean
- There is no correlation between the residuals and the predictive variables
- There is no significant autocorrelation in the residuals (so no remaining time structure)

The daily models optimised for the following 31 days have some predictive values, although the R^2 values are low. However, the plots of the forecast look reasonable, given the uncertainty intervals, and so there may be some utility in these forecasts.

The big challenge in implementing modelling using daily data will be to produce daily wholesale price forecasts. The prices_summary2 data contains daily, weekly and monthly forecasts, and we have made some progress in interpolating the data using a similar approach to previous projects, for the next 31 days. But there will always need to be interpolation, and the price forecast will never be perfect, which will limit the accuracy of the forecasts. There is then a question as to whether there should be something like the wholesale simulations for the daily forecasts. This depends on the use case, but it may be sufficient to use the intervals in the Prophet model itself.

Daily forecasting also needs a prediction of MERRA/renewables. The MERRA files which contain the scenarios are given hourly, and progress has been made in creating a daily prediction but needs more work to deal with the future capacities.

Other methods of forecasting could also be tried on the daily data. Work using LSTMs is covered in WP4. ARIMA/GARCH techniques as used the wholesale priced modelling (see separate report) could be applied here. Other methods from financial modelling could also be used, e.g as in Orlando et al[2].
Dynamic time warping has sometimes been used (including in this paper) as a metric for comparing time series. This is a metric for measuring similarity between two time series which may vary in speed or exact placement of features. We could potentially look at this to create better metrics, although it is not obvious that it solves a particular problem in the balancing costs forecasts. Dynamic time warping would be particularly useful when there is an appropriate pattern of peaks, but there is a distortion, so they are happening at the wrong time. However, if some of the techniques in Orlando et al were to be applied, then it would be worth consideration.

WP4 – Exploration of Neural Network models on Daily Data

In this work package, we explored different Neural Network approaches such as Recurrent Neural Networks in creating forecast model on daily data for 31 days.

The full details of the work can be found in the work package report WP4_30_9_23.pdf.

Data and metrics

Data

We used two versions of splits (as shown in Table 4able 5 and Table 6) on daily data to create RNN forecast models.

Data	From Date	To Date
Training	2016-08-01	2021-03-31
Validation	2021-04-01	2022-03-31
Test	2022-04-01	2022-10-31
New Test (For evaluation)	2022-11-01	2023-05-31

Table 5: Split_1 dataset

Data	From Date	To Date
Training	2016-08-01	2021-10-31
Validation	2021-11-01	2022-10-31
Test	2022-11-01	2023-05-31

[2] Orlando G et al, Financial markets’ deterministic aspects modelled by a low-dimensional equation Sci Rep. 2022; 12: 1693. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8807815/

New Test (For evaluation)	2022-11-01	2023-05-31

Table 6: Split_2 dataset

Above tables Table 5 and Table 6 show the different versions of datasets for training RNN models. We can see the trends of data for each split from Figure 10.

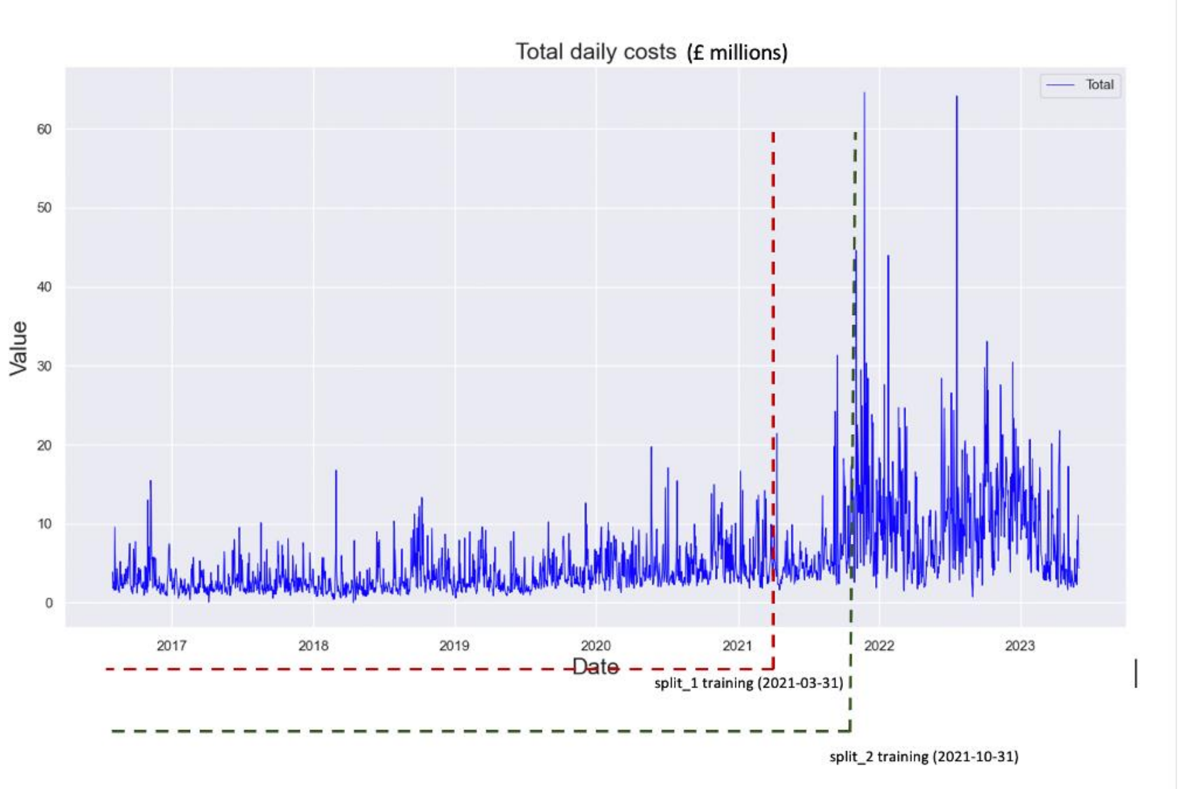


Figure 10: Total daily costs (£ millions) and splits of dataset for training

Metrics

To evaluate performance of trained models we can use different metrics, such as mean absolute error (MAE), root mean squared error (RMSE), or mean absolute percentage error (MAPE), to measure the accuracy of model.

1. For training RNN, we used **mean squared error (MSE)** as a loss function. The main advantage for using MSE is that it squares the error, which results in larger errors being minimised.
2. We have evaluated model based on the results of **mean absolute error (MAE).** Mean Absolute Error (MAE) is a measure of the average size of the mistakes in a collection of predictions, without taking their direction into account.

Feature Selection/Importance

We started with experimenting different LSTM (on split_1 dataset) forecast models to know which input features contributed more, for getting better results.

Input Features	Output Features	Model	Input Time Frame length	MAE
Price, Renewable_prop, Energy_Imbalance, Positive_Reserve, Negative_Reserve, ROCOF, Constraints_other, Total	Total	LSTM	10	0.4905

Table 7: Input features and MAE results of better LSTM

After feature engineering, the LSTM model with the above input features gave better MAE results compared to other subsets of features.

RNN experiments

We used the above list of features to create/train a multivariate many-many LSTM/GRU model with target output frame length of 31 along with Bayesian Optimisation.

Table 8 and Table 9 show the evaluation results (cumulative over 31 days) of optimised and regularised LSTM models trained on split_1, split_2 versions of training data.

Network	Input time length (Days of Input Data)	Output time length (Days of Output Data)	LSTM Units	Dense Units	Metrics	Validation	Test	New Test
LSTM	9	31	39	14	MAE	**3.834**	**4.401**	**3.771**
					R^2 SCORE	0.146	0.014	0.071

Table 8: Evaluation results of Optimised LSTM model (split_1)

Network	Input time length (Days of Input Data)	Output time length (Days of Output Data)	LSTM Units	Dense Units	Metrics	Validation	Test	New Test
LSTM	39	31	14	45	MAE	**4.937**	**3.553**	**3.553**
					R^2 SCORE	-0.130	0.029	0.029

Table 9: Evaluation results of Optimised LSTM model (split_2) (Test and New Test periods are same)

The **MAE** results (Table 8 and Table 9) of new test data (ref: Data and Metrics) for split_1 and split_2 trained models, split_2 model showed lower error **(3.553)** compared to split_1 trained model error **(3.771).** This might be because, split_2 version of LSTM has captured some extra data (compared to split_1 data) (Ref: Data and Metrics) that has similar kind of volatility as new test dataset.

Evaluation results of optimized LSTM model with test data for each day of 31 days predicted sequence

We used optimised LSTM models to evaluate the results of MAE for each day of 31-day predicted sequence on new test dataset.

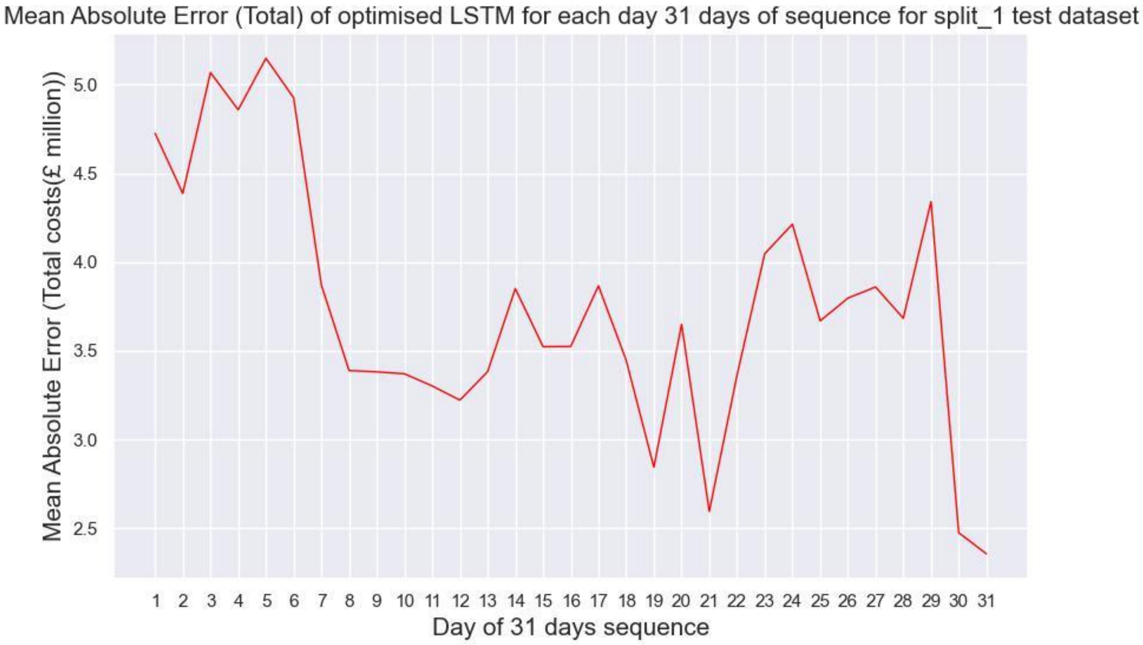


Figure 11: Mean Absolute Error (Total) of optimised LSTM for each day of 31 days of sequence (split_1)

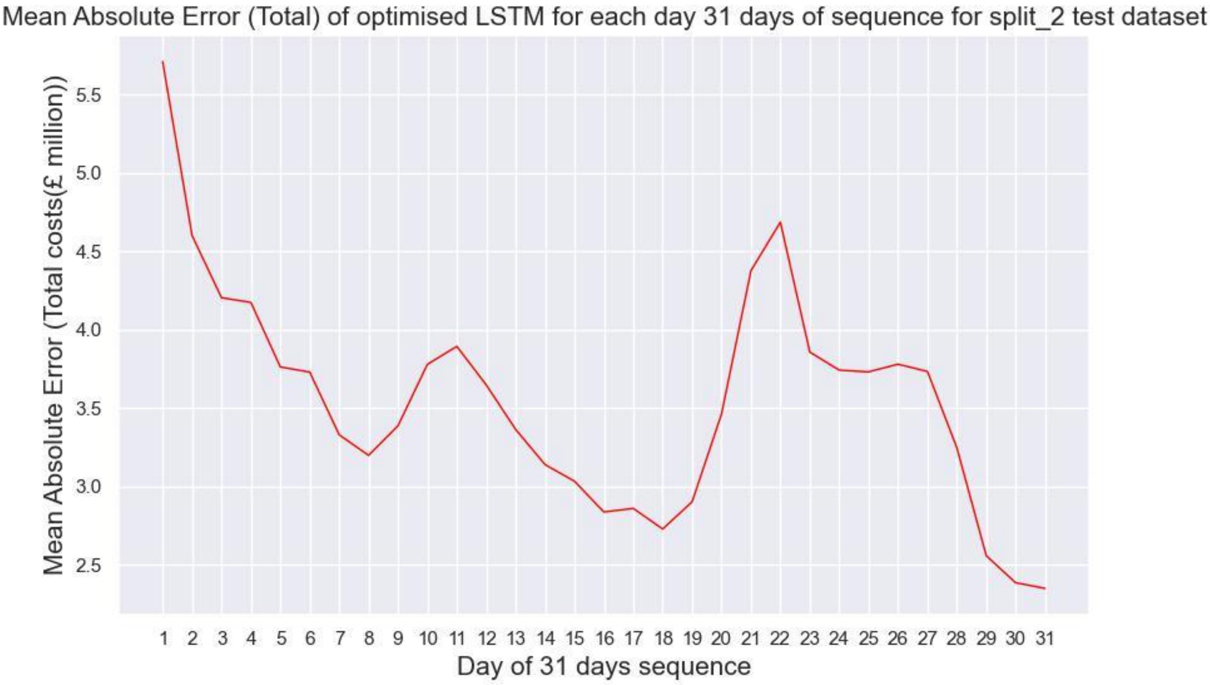


Figure 12: Mean Absolute Error (Total) of optimised LSTM for each day of 31 days of sequence (split_2)

From the results of error for each day of sequence (form figures 11, 12), there is no correlation between Error and the day number of a predicted sequence. This is because, there might be high volume of high volatility of data exist for certain kinds of sequences where we got high MAE and low volume of high volatility data exist for certain kinds of sequences.

Evaluation results of optimized LSTM model (for further analysis)

We created different plots to evaluate LSTM models, to analyse the performance of LSTM models at different time steps.

Line chart timeseries of Actuals, Forecasts, and error (X-axis daily dates, y axis costs) first day of each 31-day sequence

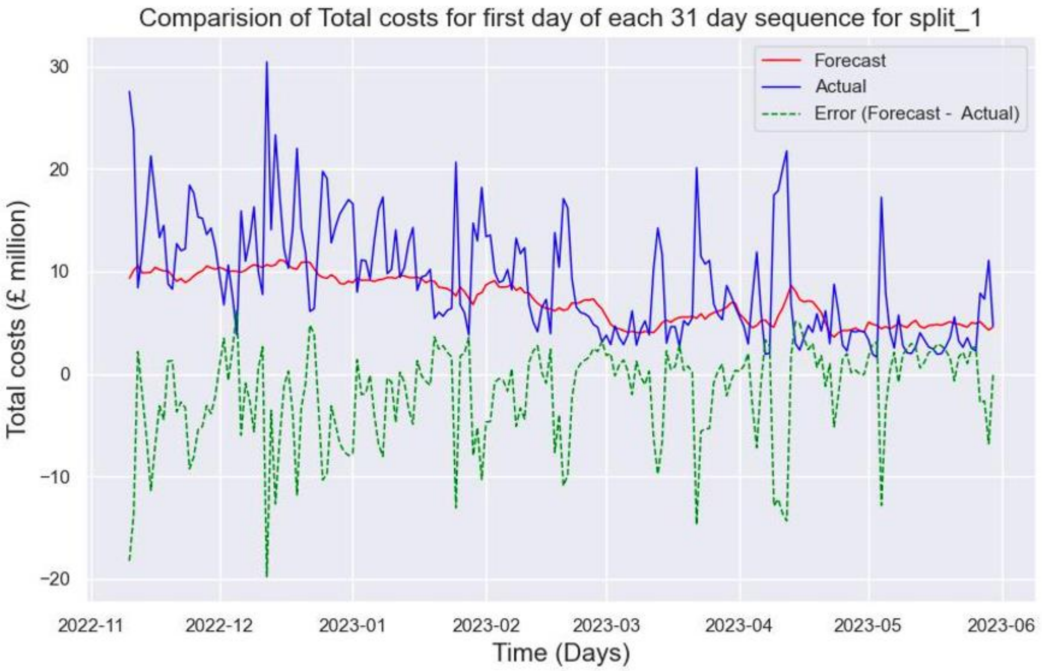


Figure 13: Line chart timeseries for first of each of 31-day sequence for split_1

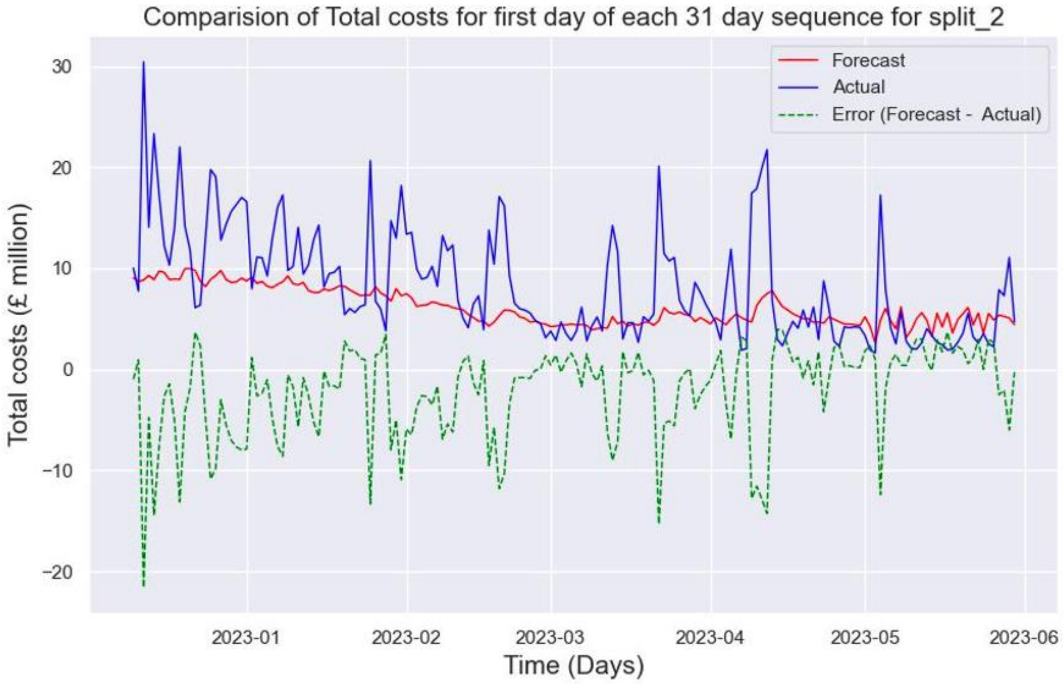


Figure 14: Line chart timeseries for first of each of 31-day sequence for split_2

From the results of Figure 13 and Figure 14, the forecast models (split_1, split_2) do not capture the volatility seen in the actuals, and the forecast sits towards the bottom of the actual cost range. This means the errors are mostly negative. However, the forecast does capture the high-level trend without a notable lag.

Discussion and future work for WP4

To explore more about the behaviour of LSTM models, we created various reports and plots for further analysis. We plotted line charts of forecasts, actual and errors to show the trends of forecasts for the first day of each predicted sequence, and similarly for the forecasts for each month together. We created various histograms and boxplots to analyse the distribution of errors and skewness for predictions for each day of the week, and for each day difference of target and forecast dates. We also created scatter plots used to analyse relation between error with Price and Renewable_prop to extra how we can use the regression line to create new LSTM model. The exploratory plots are available in the detailed report WP4_30_9_23.pdf.

The future work could be extending these experiments to new version of splits training data with high volatility to check the behaviour LSTM learning curve. Hyperparameters and regularization parameters may likely change when we add more data to our training set if the added data is new or unseen by your model before. The Transformer model can be used to address forecasting in time series analysis. These type of deep learning models used for natural language processing and computer vision tasks. They utilize a mechanism called “self-attention” to process sequential input data.

Overall discussion and future work

A modest improvement was obtained in the monthly forecast by using Prophet to predict the balancing costs, from 78 to 73 MAE in total costs, based on using actual values for forecasting. NGESO need to consider whether this is sufficient improvement to merit changing the forecasting method.

Daily forecasts have been produced for the first time, up to 31 days ahead. Forecasting total daily costs, using similar machine learning methods to the monthly data gives an **MAE of 3.23** cumulatively over 31 days when forecast for each possible 31-day period in the test set (1/11/23 - 30/4/23). Using neural network based LSTM methods, the best model, trained on the split_2 data and forecasting on the same test data, gives an **MAE of 3.55**.

The accuracy of forecasts has been hampered by the volatility of wholesale prices during 2021 onwards. The timings are such that we end up training a model on data before this period, and then validating it on data after this period. Machine learning models do not generalise well when training and validation data comes from different distributions. We have explored new ways of creating simulations for wholesale data, but as these are always going to be statistical simulations rather than a firm forecast, there is always going to be uncertainty in the final costs. This is partially taken into account by the existing simulation-based approach, but they still cluster fairly tightly around the central forecast. Using alternative modelling strategies for wholesale prices may help cover a broader range of possibilities.

Two versions of splits of daily data were used to train different versions of forecast LSTM models with the view to creating alternative models to improve the results. Table 10 shows the results of the forecast model with comparison to results of Prophet and ARIMA.

Work package	Model	MAE (cumulative over 31 days)	MAE (spot – day 31)	R^2 (cumulative over 31 days)	R^2 (spot – day 31)	MAPE (cumulative over 31 days)	MAPE (spot – day 31)
WP3	Linear + ARIMA	3.28	3.07	0.41	0.46	0.52	0.57
	Prophet model 6 hyperparameters	3.23	3.43	0.43	0.31	0.51	0.58
WP4	LSTM (Split_1)	3.77		0.156		0.522	
	LSTM (Split_2)	3.55		0.029		0.497	

Table 10: Results of forecast models

Of the two versions of the forecasting models of WP4, the LSTM (split_2) gave a better MAE (cumulative over 31 days of new test data) of 3.55, but did not succeed in giving better results compared to the models of WP3. But when we compare results of both LSTM models (split_1, split_2), we could say split_2 was the best model, as it was trained on extra data that has more volatility compared to split_1 training dataset. The overall robustness of ARIMA and Prophet are better than that of LSTM and are more adaptable to the existing dataset. ARIMA, Prophet has more stable performance under the lack of data volume, while LSTMs could have better performance with high data volume.

LSTM models, being neural network based, are particularly dependent on having a large volume of data, and this may be one reason why they have performed less well than the Prophet based model here. As more “high volatility” data comes in, or prices start to settle, we would expect the forecasting performance to improve, and it would be worth retraining the LSTM models (the Prophet models take everything in the past as their training data each time). We may already be starting to see improvements in the test data, which gives a better MAE than the validation set for Prophet models.

The two daily forecasting methods are also subtly different in how they would be implemented. The Prophet based method requires something very similar to the monthly pipeline. This means that a version of the wholesale simulations for daily data is required, and forecast values of the other variable are required daily. On the other hand, the LSTM based method does not require values for the period being forecast but needs daily data for the days leading up to it, including for actual costs. This means that the decision on whether to proceed with one of these daily forecasting methods, and which one, depends on not just on their accuracy but also on the availability of the data required to make the forecast. One approach would be to start with the Prophet model and revisit the LSTM model once more data is available.

We found that some extra variables relating to renewables and demand improved performance, but those from very different data (margin and inertia) were less helpful, and would also have been difficult to forecast, so were not ultimately used.

In the investigations of the importance of different variables, which focussed on the Constraints component, it was found that renewable proportion had a greater influence than wholesale prices. It could therefore be helpful in future to look in more depth at how this is forecast, maybe using something similar to the wholesale simulation approach.

Before adopting the new approach to monthly forecasting, it would be a good idea to run the new forecast in parallel to the old one for a few months to see how it performs under real world conditions.

Ultimately, it has proved very difficult to get a good forecast, which may be in part due to market sentiment and behavioural aspects driving the costs as much as underlying driving factors. It may be worth treating the costs in a similar way to prices on financial markets and applying models like GARCH directly to the costs.