Public

January 2026

CrowdFlex: Beta

Modelling: Final Report

NESO
National Energy System Operator

Public

Contents

Executive summary

CrowdFlex is NESO-led innovation project, funded by Ofgem's Strategic Innovation Fund (SIF), which is investigating the potential of domestic flexibility to help operate the grid. CrowdFlex is aiming to establish domestic flexibility as a reliable energy and grid management resource by identifying the technology capability, understanding the statistical nature of flexibility, and aligning NESO and DSO requirements. Through large-scale randomised control consumer trials, CrowdFlex is collecting data to develop demand and consumer flexibility prediction models using common APIs.

NESO is delivering CrowdFlex with a consortium of industry partners: OVO, Ohme, Centre for Net Zero, ERM, AWS, National Grid Electricity Distribution, Scottish and Southern Electricity Networks, and supported by Smart Grid Consultancy, CGI, Smith Institute and Centre for Sustainable Energy.

CrowdFlex has successfully delivered two state-of-the-art probabilistic forecasting models: the Available Flexibility Model (AFM) and the Expected Delivery Model (EDM). These models have been trained on wide scale consumer trial data, and are supported by robust data pipelines and an intuitive web-based user interface (UI). These tools are poised for integration into NESO's core operations, enabling real-time, data driven decision making for grid management.

Key findings

- **Domestic flexibility is predictable**: The AFM and EDM both provide predictions of the range of potential flexibility delivery levels and perform better than a naïve forecast for almost all combinations of target quantile for turn-up and turn-down events. These results demonstrate that it is possible to model domestic flexibility probabilistically using these methods and data, in preparation for BAU.
- **Key factors impacting flex forecasts identified**: Delivered flexibility varies strongly with forecast demand, geographic location, temperature, flexibility targets and temporal factors (hour of day, day of week etc.) consistently having high feature importance for all models, with target flexibility also appearing for the EDM. These insights can be used to plan future data collection by those managing flex services and to target improvements to data quality.
- **Large amounts of new data collected:** Data was collected across 609 events with **98.81%** data retention representing flexibility delivery from over 100,000 participants across 330 GB grid supply points. Through this data collection, CrowdFlex has set a new standard for collecting and ensuring quality in a large volume of data from multiple sources which demonstrates strong potential for applicability to BAU.
- **Flexible models for changing circumstances:** The AFM and EDM models show promise for integration into NESO's grid operations, where future iterations of the models will need to be tailored to the changing energy landscape. This report shows that flexibility predictions can be made using different model architectures and that models can be trained for bespoke use cases such as custom regions using the same data, demonstrating the models are well placed for future operations as they evolve.

CrowdFlex demonstrates the transformative potential for domestic flexibility to support grid management and decarbonisation, and provides a template for future data-driven innovation in the energy sector. The project's technical advances, insights, and lessons provide a strong foundation for future work and successful BAU integration within NESO's operations to increase use of domestic flexibility for grid management in support of achieving the Government's Clean Power 2030 Action Plan (CP30).

Glossary

Term	Definition
AAE	Advanced Analytics Environment
Actuals	Actual energy used by all households in a GSP during a half-hour settlement period. Measured after the event has concluded and data has been collected from smart meters.
ADF	Azure Data Factory
AFM	Available Flexibility Model
ALZ	Azure Landing Zone
Anti/antisymmetric event	Flex event requesting turn-up to the north of the SCOTEX boundary (essentially Scotland) and turn-down to the south.
API	Application programming interface
Availability	CrowdFlex event type where consumers are encouraged to plug in their EV for extended periods so that DSRSPs can shift when charging occurs (either into or out of the event).
Azure ML	Azure Machine Learning
BAU	Business as usual
Baseline	Expected energy usage had the flexibility event not occurred. This is estimated by the DSRSPs, either using a control group (OVO availability) or using the industry-standard P376 (all other events).
Critical down	A set of eight turn-down utilisation events in winter 2024 for which consumers were offered larger incentives to turn down their energy usage.
CP30	Clean Power 2030 Action Plan. The UK Government's action plan that sets out a pathway to a clean power system by 2030.
CNZ	Centre for Net Zero
Delivered flexibility	The change in energy usage by consumers. Calculated by subtracting the baseline from the actuals (see Figure 2). Sometimes abbreviated to 'flexibility' or just 'flex'.

Demand forecast	DSRSP-provided data giving predictions of future energy usage by consumers assuming no flexibility event is taking place (i.e. a counterfactual). This is often referred to simply as 'forecast' data or 'DSRSP forecasts' (the latter to distinguish it from weather).
Demand shift	Consumers moving their energy use in response to a flexibility event. For example, consumers may move when they would normally use an appliance and increase their energy demands before or after an event so as to turn-down their usage within the event period.
DFS	Demand Flexibility Service. A nationwide flexibility service providing turn-down response. We have also defined a model region 'DFS', named after this service, to predict for all GSPs in Great Britain. Quotes are always used when referring to the model region.
DSO	Distribution system operator
DSRSP	Demand-side response service provider. In this report, we deal with two DSRSPs – the energy providers Ohme and OVO.
EDM	Expected Delivery Model
EV	Electric vehicle
GB	Great Britain
GBT	Gradient-boosted tree. The model architecture used in the latest AFM and EDM.
GSP	Grid supply point. The connection between transmission and distribution networks. When referring to GSP in this report, we typically refer to the usual area served by that GSP. We have also defined a model region 'GSP' to predict for all GSPs individually. Quotes are always used when referring to the model region.
GUI	Graphical user interface
KPI	Key performance indicator
LCM	Local Constraint Market. A flexibility service designed to ease constraints across the B6 boundary in Scotland. The included region varies by date, although typically includes most GSPs in Scotland. We have also defined a model region 'LCM', named after this service, to predict for all GSPs that were included in LCM at any point during 2024. Quotes are always used when referring to the model region.

LQR	Linear quantile regression
Model region	Set of GSPs to predict for and how to aggregate them when modelling. Region names are always in quotes: 'DFS', 'LCM' or 'GSP'.
MPAN	Metering point administration number. Used here to refer to the granularity of readings collected for energy usage by one household.
Naïve forecast	Value of corresponding quantile in the training data in the selected flex direction, and optionally filtered by in event/shoulder (EDM only), with all other input parameters ignored. Only used when calculating SQL. Not to be confused with the naïve forecast typically used when calculating MASE (mean absolute scaled error) of time series forecasts, nor to be used as a benchmark when comparing alternative methods.
NESO	National Energy System Operator
PN	Physical Notification, used in Balancing System.
Quantile	Probabilistic estimate of delivered flexibility. For example, given a fixed set of inputs, we expect 60% of observations to be less than or equal to the 0.60 quantile. When evaluating risk, this means we can be 40% sure of getting a flexibility response greater than the 0.60 quantile.
Quantile loss	A measurement of the distance between the predicted quantile and the true value of the quantile. It is analogous to mean absolute error (and identical for the median prediction) but with each error weighted unequally based on the predicted quantile.
Settlement period	Half-hour interval for settlement in energy markets.
SGC	Smart Grid Consultancy. A CrowdFlex consortium member.
Shoulder	Period of time either side of a flexibility event (six hours before and after). Demand forecasts, baselines and actuals were collected during the shoulder period in the summer 2025 trial for use in training the EDM to model demand shift.
SIF	Strategic Infrastructure Fund
SQL	Scaled quantile loss. A measurement of the distance between the predicted quantile and the true value, scaled by the distance for the naïve forecast. Values less than one indicate the model is performing better than the naïve forecast.

Target flexibility	Specific amount of flexibility required from an event and aimed for when delivering demand flexibility.
UI	User interface. Primarily referring to the dashboard we constructed to query our models and display the resulting predictions.
Utilisation	CrowdFlex event type where users flex the energy usage of their entire household up or down.

1. Introduction

1.1 Project background and context

CrowdFlex is a NESO project that aims to establish domestic flexibility as a reliable energy and grid management resource. The project explores how domestic flexibility can be used in grid operations to support an affordable, secure, net-zero energy system.

CrowdFlex key deliverables are:

- Probabilistic models to empower NESO forecasting of domestic demand and flexibility.
- Large-scale consumer trials that enable model development and increased understanding of the technical capabilities of the different forms of domestic flexibility.
- Mapping the characteristics of the different forms of domestic flexibility to existing NESO services to establish a pathway to rapidly accelerate domestic flexibility to NESO BAU.

Through these activities, CrowdFlex enables NESO and distribution system operators (DSOs) to fully utilise domestic flexibility. The approach intends to reduce operational costs and offer a reliable option to costly capacity and network reinforcement investments. Ultimately, this will contribute to lowering consumer energy bills and accelerating decarbonisation across the whole energy system.

1.2 Project objectives

The model workstream for CrowdFlex has designed, built and developed two models, the Available Flexibility Model (AFM) and Expected Delivery Model (EDM). Both models work together to provide performant models of flexibility potential and response, are informed by large-scale consumer trials, and will support NESO to establish domestic flexibility as a business-as-usual (BAU) grid management resource.

Each model is further described in Table 1.

Model	Description and purpose
Available Flexibility Model **AFM**	**Purpose:** Inform NESO how much domestic flexibility might be available and its location and timeframe. **Overview:** Forecast the distribution of total flexibility that could be delivered across all flexibility markets, conditional on NESO choosing to dispatch some domestic flexibility, i.e. available flexibility.
Expected Delivery Model **EDM**	**Purpose:** Inform NESO of the reliability of domestic flexibility dispatched for a specific event. **Overview:** Forecast the distribution of delivered flexibility conditional on a specific dispatch, i.e. the specific volumes procured from each DSRSP. The model will also forecast the impact of dispatched flexibility on demand outside the flexibility period (demand shift or creation/destruction).

Table 1: AFM and EDM descriptions.

To deliver these models and enable NESO to use them, Smith Institute designed, built and developed the surrounding infrastructure:

- Data pipelines to ingest demand-side response service provider (DSRSP) consumer trial demand data via an API.
- Code to train the AFM and EDM, generate flexibility forecasts, and evaluate their performance.
- A user interface (UI) for NESO users to interact with the model, query predictions, and interpret the models' findings in the form of a web-based dashboard.

The consumer trials were conducted by OVO Energy, Ohme EV and Centre for Net Zero over a series of summer and winter trials covering both behaviour around plugging in and charging EVs and flexibility in the energy used by entire households. The trial types and the data collected for each are detailed in 2.1 Data sources and consortium contributions. The design and scheduling of these trials was developed by multiple consortium partners together, with Smith Institute consulted on the requirements and implications for training the models.

Figure 1 below highlights the different parts of the system where the models get their data to enable domestic flexibility. Demand forecasts and post-event actuals were supplied by the DSRSPs via APIs and these were combined with Met Office weather forecasts in data pipelines designed to supply data to the models as a live service.

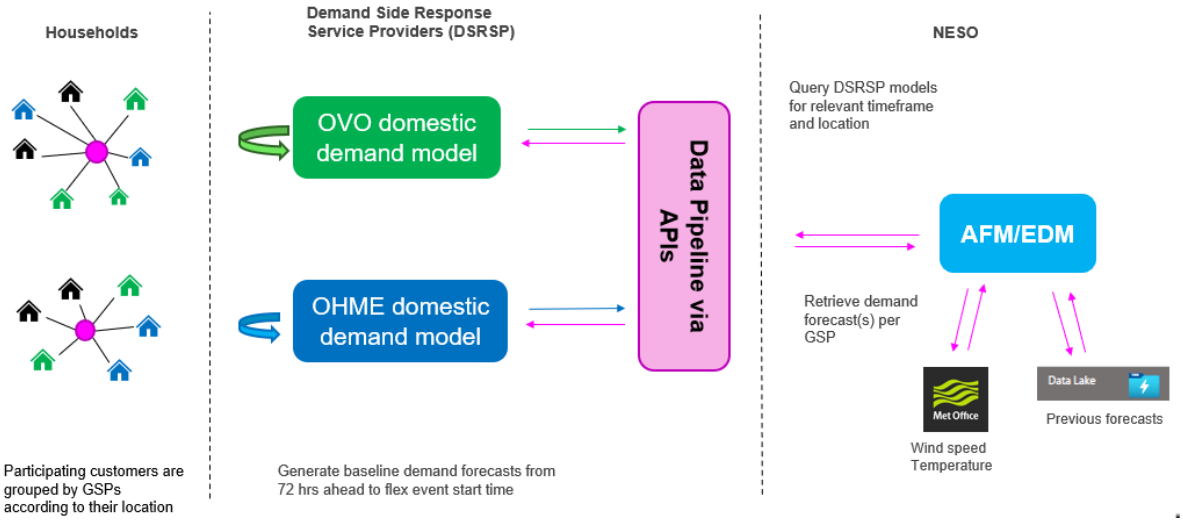

Figure 1: Where the data for the models come from.

Using demand and weather data, the AFM and EDM provide probabilistic forecasts for available and expected flexibility at a Grid Supply Point level across Great Britain. The probabilistic nature of the forecasts allows NESO to not only see the most likely outcome, but the range of possible delivered flexibilities. They do this by providing a set of quantiles that characterise the delivered flexibility that can be expected in cases ranging from 95% likely (0.05 quantile) to only 5% likely (0.95 quantile). Thus, risks and opportunities can be quantified by NESO users and actions taken as appropriate.

Engagement with relevant teams across NESO was carried out to develop use cases for the models. The engagement with the relevant teams about the potential use cases for the domestic flexibility models included discussion of their current processes around domestic flexibility and constraint management. The engagement also included how domestic flexibility services could evolve in the future and to understand the potential suitability of the AFM and EDM to support in BAU. Further information can be found in the implementation strategy document.

In this report we describe the various aspects of work undertaken during this project. We explore the supplied data and data processing pipelines we developed. We summarise the modelling performed and share results of the latest model evaluations. We describe the functionality of the user interface used to request flexibility forecasts. Finally, we share details of the challenges we faced and overcame during this project and the learnings that can be taken from them. Further technical details relevant for ongoing maintenance and development can be found in the appendices.

2. Data overview

2.1 Data sources and consortium contributions

The CrowdFlex beta trials drew together data from NESO, OVO, Ohme, Met Office and Ordnance Survey. There were three sets of trials, which took place May-July 2024 (summer 2024), September 2024-April 2025 (winter 2024), and July-October 2025 (summer 2025).

There were a mixture of availability and utilisation events. For availability events, participants are encouraged to plug in their EVs for extended periods of time so that their DSRSP can shift when charging occurs according to the requirements of the event. The data here is collected at the level of individual charging assets. In utilisation events, households are incentivised to flex their entire domestic energy usage up or down. OVO[1] recorded this data at the metering point administration number (MPAN) level. DSRSPs supplied both availability and utilisation trial data to Smith Institute aggregated at grid supply point (GSP) level. The majority of data provided was for time points within events. However, for use by the EDM, Ohme and OVO also supplied 'shoulder' data (data six hours either side of each event) as part of the summer 2025 trial.

The majority of flexibility events are either 'turn-up' or 'turn-down', where all participants across the country received an incentive to flex their usage in the same direction. There were a small number of 'antisymmetic' events (around 6% in total, all from the summer 2024 and winter 2024 trials). In these, consumers north of the SCOTEX boundary were incentivised to turn-up their usage, whilst those south of the boundary were incentivised to turn-down their usage. In practice, this boundary is very close to the England/Scotland border.

We developed and deployed a structured set of data pipelines to collect, validate, collate and store the trial demand data and Met Office weather data used by the models. These are described further in Section 2.2.

DSRSP forecasts and actuals

The demand-side response service providers (DSRSPs) provided demand forecasts, baseline demand, and actual energy consumption for each settlement period associated with an event. For summer 2024 and winter 2024, these were just the settlement periods during an event. For summer 2025, the windows were extended to include six hours either side of the event, as required by the EDM. The consortium chose P376[2] as the baseline for all trials except OVO availability, which used a control group. OVO opted to use a control group because the high frequency of availability events limited the applicability of P376, which relies on a recent history of event-free days. These baselining methods estimate the amount of energy that would have been

[1] Ohme did not take part in the utilisation trials.

[2] https://www.ofgem.gov.uk/sites/default/files/2021-08/Approval_of_BSCMod_P376162807597130O.pdf

used by the trial participants if no flexibility event had occurred. This then allows us to calculate the flexibility delivered for each settlement period and grid supply point (GSP) as the actual usage minus the baseline – this is illustrated in Figure 2.

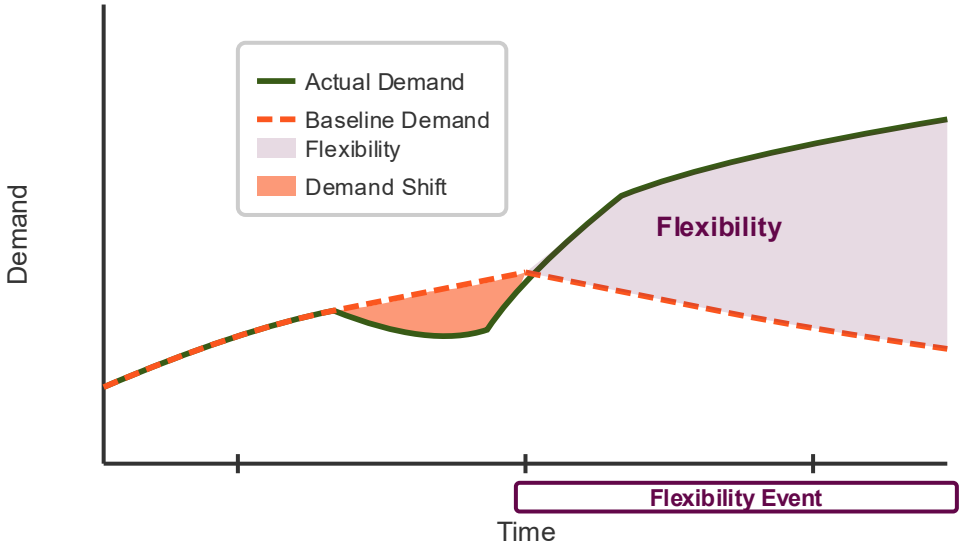

Figure 2: Illustration of how (delivered) flexibility is defined. Baseline demand is subtracted from the actual energy usage. Only settlement periods inside the flex event are counted.

For most of the trials period, OVO also had forecast demand data available continuously (not just during flex events). This was not required, and therefore was not available from Ohme. We did not make use of it except where flex events were rescheduled or event times configured incorrectly.

Each DSRSP independently created an application programming interface (API) for sharing their data, according to the CrowdFlex API Specification (D1.2). The specification includes separate paths for their demand forecasts ('forecast') and observed consumption/baselines ('actuals'), so that each can be updated and queried on independent schedules. We originally intended to query demand forecasts hourly to ensure end users always have the most up-to-date information when planning flexibility events, although we later cut the frequency to daily to match the daily forecast cadence of the DSRSPs, thereby saving compute costs during the trials. Each collected forecast has a different 'lead time', representing the lag between its creation and the time it is predicting for. Actuals are queried around one week after the end of the corresponding flex event to allow time for all the relevant data to be collected by the DSRSPs and processed for us.

Our data pipelines remotely request data from the APIs by encoding required parameters, such as the event start time, within the API URL. If the request is successful, the API returns the requested data, which is ingested into our data pipelines. The data contained within is at the level

of individual GSPs (or GSP groups – see GSP aggregations) with measurements in kWh for each settlement period (labelled by its start time). Formats are uniform across DSRSPs to enable future expansion to more DSRSPs.

Spatial aggregation

To enable DSRSPs to aggregate trial data at GSP level, we assigned households to GSPs as part of our original preparations for data collection. We matched shapefiles from NESO[3] describing the area covered by each GSP with postcode data from Ordnance Survey Code-Point to map each postcode to a GSP. The DSRSPs assigned their customers to GSPs using our mapping. This process missed a small number of postcodes, which we assigned manually to ensure we were able to collect data from all participating households.

To comply with GDPR legislation, GSPs had to have sufficiently many participants to ensure anonymity. Thus, we agreed that DSRSPs would group some GSPs prior to releasing their data. For OVO, GSPs with fewer than ten participants for utilisation trials and five participants for availability trials were combined with neighbouring GSPs until the whole group was above this threshold. The grouping prioritised the smallest neighbours (by participant count). For Ohme, GSPs with fewer than five participants were dropped (as Ohme did not group them). The grouping process was carried out by Smith Institute for each DSRSP and trial and was needed in advance of collecting data.

When processing data in our pipelines, we split these grouped GSPs back into their constituent parts using a weighted division calculated from the participant counts as they were when the GSP grouping was agreed. This was to ensure full future compatibility in the event that GSP participant counts and groupings changed over time. Indeed, the DSRSPs later reported that the participant counts changed between events and even within an event (e.g. if a meter connection failed). However, we did not receive these dynamic counts and considered the number of participants in a GSP to be fixed within a given trial.

In BAU, we expect all GSPs to contain sufficiently many participants to not require any grouping or else for NESO to have sufficient safeguards, possibly in the form of a data sharing agreement.

Data quality and quantity

Following the completion of the final set of trials, we have collected data for up to 330 individual GSPs across 609 flexibility events. We compared the collected data in our training datasets (which combine weather, demand forecasts, and actuals) to our expectations from scheduled events. Here, we count each DSRSP and trial type separately when calculating the number of settlement periods we expect.

[3] https://www.neso.energy/data-portal/gis-boundaries-gb-grid-supply-points/gsp_regions_20220314_esri_shapefile

We give figures for the EDM, which include shoulder periods for the summer 2025 event, and match the AFM data for the summer and winter 2024 trials. The resulting figures are very positive:

- **98.81%** (5,812,503/5,882,409) settlement period/trial/DSRSP/GSP/lead time combinations are present out of those expected.
  We consider this figure to be the headline statistic/KPI on data completeness.
- **95.04%** (1,863,503/1,960,803) settlement period/trial/DSRSP/GSP combinations for at least some lead times.
- **98.98%** (6,667/6,736) settlement period/trial/DSRSP combinations have data for at least some GSPs/lead times.

To get the 'expected' figures (the fraction denominators), we make the following assumptions:

- Each data point (when counted for the first statistic) should be present for three lead times – since demand forecasts are updated daily and we collect them over a 72-hour period[4].
- The maximum number of expected GSPs for each trial/DSRSP is equal to the maximum number observed across all data for that trial and DSRSP. For example, across the winter 2024 trial, we received from Ohme availability data for 237 distinct GSPs. Thus, we assume that any data from them for winter 2024 should be present for each of those 237.

The 5.8m row count is less than the actual amount of data in our gold training table (which currently has around 12.5m rows) and the silver forecast table (which currently has around 21.5m rows)[5]. The excess comes from:

- MPAN-level availability data from OVO, which was collected for potential future use and compatibility with the utilisation trial but has not been used for model training.
- Hourly collection of demand forecasts (from OVO) in earlier trials. Wholly duplicated rows are not present in the training data, but the different cadence of weather collection means we get rows with the same demand forecasts but updated weather information. Since this does not represent genuinely new data, we reduced to collecting daily. Where hourly data has since been repulled to improve quality, we only sent backfill queries to simulate a daily rate.

The high rate of data completeness was achieved through close collaboration with the DSRSPs, particularly when it came to diagnosing and addressing quality issues whilst the trials were still ongoing. We encountered completeness issues early in each trial but have since resolved all

[4] Where we collected forecasts more frequently than daily (and where the advance period intersects with the EDM shoulder), this isn't always true. In general, we expect 2-4 demand forecasts for each event, with the exact number dependent on when we sent our daily queries and when each DSRSP updated their forecasts.

[5] See 2.2 Data preprocessing and integration for descriptions of various tables.

those which pertained to notable quantities of data (some small amount of attrition is to be expected when collecting data in these volumes).

There are a couple of factors that still affect the quality of the data that has been used to train the models. These are either intrinsic to the methodology or cannot be fixed retroactively:

1. Baseline accuracy

The P376 baseline (used in all trials except OVO availability, which used a control group) has several well-known limitations[6]. In particular, when there are frequent flexibility events, the time window used for calculating the P376 baseline potentially becomes very large and unrepresentative of the event. We observe such inaccuracies as seasonal drifts in the data (discussed further below in our exploratory data analysis). We have not analysed in depth the implications of using P376 versus a control group has for modelling multiple DSRSPs together, nor which method we expect to produce the most accurate estimates of delivered flexibility.

2. DSRSP hardware issues

Ohme reported earlier problems where a firmware fault in some metering equipment resulted in occasional anomalous readings. This showed up in our pipelines as negative baselines/actuals values. Although this happened infrequently, it nevertheless becomes noticeable and problematic when data is collected for ~10,000 customers. Ohme successfully addressed the root cause in time for the summer 2025 trial. Unfortunately, the only fix that can be applied to historic data is to set the negative values to zero. This means some unknown number of GSPs have data for fewer than expected households, mostly for data collected in the first two trials. We opted to treat this just as a source of noise in the data that the model uses for training.

Event schedule

For each trial, the consortium constructed an event schedule prior to the start of the trial, detailing the parameters for each flex event. Centre for Net Zero (CNZ) lead the process for summer 2024 and summer 2025, and Smart Grid Consultancy (SGC) lead for winter 2024. Although the DSRSPs saw the schedule well in advance, they did not insert the schedule into their optimisers or start to dispatch customers prior to the DSRSPs' notice period for that event, in order to simulate BAU.

A spreadsheet contains all schedule parameters, from which we (summer 2024 and winter 2024) or SGC (summer 2025) extracted the parameters required for the modelling workstream as a CSV file: event ID, trial type (availability or utilisation), start time, duration and flex direction (turn-up, turn-down or anti). We then processed this CSV into the format expected by our data pipelines. Note that the full event schedule contains parameters not used for modelling such as

[6] https://www.neso.energy/document/279586/download

notice period and the choice of incentives to be offered to customers. NESO published the relevant parts of the event schedule to DSRSPs via an API[7], as part of their BAU simulation.

On a few occasions, NESO revised the event schedule at short notice due to unforeseen circumstances, such as windy conditions placing large constraints on the grid, which would have been unduly exacerbated by the scheduled event. For these, we reprocessed the event schedule to reconfigure our system.

Similarly, sometimes a single DSRSP failed to dispatch, a DSRSP dispatched at the wrong time due to a technical fault, or we missed a NESO update to the schedule at very short notice. In these cases, DSRSPs informed us directly of the change. The timescales involved meant we typically identified missing data before they were able to inform us of changes. This is an expected feature of the data pipelines, since the automated validation checks give feedback quickly. We then manually edited our copy of the event schedule and repulled the data - see 4.1 Data-related challenges for a discussion of backfilling DSRSP data.

Furthermore, all customers in some GSPs were not dispatched for some events, e.g. on request by the affected distribution system operator (DSO). These are not reflected in the event schedule. Instead, the DSRSPs omitted these GSPs from their actual delivery data, thereby excluding it from our model training data.

Weather forecasts

We expect weather to be a significant influencer of demand flexibility. We collected forecasts from the Met Office 3-hour site-specific API[8] every three hours, using a similar process to DSRSP data.

However, due to infrastructure issues, forecasts are missing for several months in parts of summer 2024 and winter 2024. We backfilled these from a Met Office archive of their UK atmospheric forecast hosted on AWS[9]. Although this data is of the same quality, it had to be converted from a different coordinate system and set of weather stations. This may have impacted the exact values used as model input.

---

[7] The API only includes events published to DSRSPs, excluding any cancelled well in advance, and whether each DSO has accepted or rejected the event.

[8] https://datahub.metoffice.gov.uk/docs/f/category/site-specific/type/site-specific/api-documentation

[9] https://aws.amazon.com/marketplace/pp/prodview-oiodcatwsyjwm?sr=0-1&ref_=beagle&applicationId=AWSMPContessa

2.2 Data preprocessing and integration

Our data processing pipelines are built in Azure Data Factory. We used a 'medallion' structure, with the processing steps divided up into bronze, silver, and gold. Each stage carries out a different, independent function and their outputs are saved to allow flexibility over future use, and for debugging.

Bronze

The bronze pipelines are responsible for ingestion of raw data and some basic sanity checking through schema validation. The pipeline is split into 3 parts – demand forecasts and actuals from each participating DSRSP, and weather data from the Met Office. The pipelines collect data at regular intervals: daily for demand data (forecasts and actuals), 3-hourly for weather data.

Silver

The silver pipelines are responsible for converting the bronze data into a tabular form, as well as carrying out more thorough validation steps. The general idea with 'silver' data is that it is confirmed to be clean and as complete as possible, but that it also remains generically suitable for a variety of further tasks. Validation included checking the data we received for self-consistency (for example, 'does the provided data series start after the stated start time?') as well as against our expectations ('does the provided start time align with the expected flexibility event?'). We also carry out merge operations to assign internal labelling to the data such as a numerical ID for each GSP (or GSP group). The full set of checks and steps can be found in the data pipelines technical documentation.

Errors are raised by these pipelines if data fails validation, or if there are cases where data has unexpected gaps compared to what is expected from our event schedule. Some of these gaps (especially for actuals) require manual action, since they include cases where the flex event which occurred does not match our configuration. We then need to verify whether the configuration needs updating, or whether the data we have received was incorrect. In cases where data is missing, DSRSPs have sometimes been able to provide it for us at a later date, although there are also occasions where it remains permanently unavailable. In cases where a single data point is missing, it may not be worth the time required to manually requeue the corresponding forecast, especially if it is already present for other lead times.

The silver data storage contains three main tables: demand forecasts, actuals and weather data. There are additional tables for storing data which fails validation. As the project concludes, these still have some content but have been reviewed to ensure that it's either innocuous (such as some GSPs from summer 2024 with incorrectly formatted names, where the corrected data was supplied alongside rather than replacing it) or limited in volume (such as two total data points with a timestamp that isn't valid for a half-hour settlement period).

Gold

The role of the 'gold' pipelines is to take the tabular silver data and convert it into a format that is immediately ready for use by the models. There are three main tables here – one each for training the AFM and EDM, plus a third 'inference' table used for input to the deployed models. It is this third table that gets queried when a user requests predictions from the user interface (UI).

The gold inference table contains data merged from the silver demand forecast and weather tables. This is also the stage where we split apart GSP groups into the constituent GSPs. As well as giving better granularity in model predictions, this also ensures that there are no issues caused by having different groupings of GSPs between different trials/DSRSPs. The inference table retains only one year of data, since we anticipate that most users are interested in future (or very recent) predictions. In BAU, this could be reduced further to speed up the UI response times. The full training data can still be used to analyse any historical period. While we need to use the event schedule for our silver validation, the inference table should be free of any reference to flexibility events or event details (such as duration or flex direction) as they are inputs into the UI and not retrieved from the inference tables. This is to ensure likely compatibility into BAU – where events may be planned based on model predictions, rather than scheduled far in advance.

The two training tables then further merge this data with the corresponding set of actuals – we anticipate that these will always correspond to known flexibility events.

2.3 Exploratory data analysis

Summaries of data

The most expansive set of model-ready data is in the 'EDM training' gold Delta Lake. This contains within it the data in the other gold tables (they have fewer rows/columns as described above). After the conclusion of all the CrowdFlex trials, this has just over 12.5 million (12,515,677) rows. This covers the three CrowdFlex trials in summer 2024 (32 events, 7 May-12 July 2024), winter 2024 (458 events, 18 September 2024 - 4 April 2025), and summer 2025 (119 events, 7 July-3 October 2025). These are a mixture of availability trials (Ohme and OVO) and utilisation (OVO only).

We first show plots of the flex delivered per settlement period for each event, summed over GSP. These show the amounts of flexibility that were delivered in each set of trials, giving an indication of the numbers the AFM is aiming to predict. We show each DSRSP, trial type, and season on separate graphs as variations in methodology and participant count means the values on each are not all comparable. Antisymmetric events are excluded since comparing them in this way is not so insightful.

Figure 3 shows the flexibility delivered by Ohme in the summer 2024 trial. We note that the turn-down performance is typically stronger than that of the turn-up events and that three of the 'turn-up' events have a response that goes the 'wrong way'. This can happen if the baseline used to calculate the total flex is unreliable but can also be a genuine occurrence due to customer behaviour (which the AFM and EDM will need to account for). It is more likely to happen when

considering smaller numbers of participants/GSPs and, for Ohme, could also be down to the zeroed meter readings discussed above.

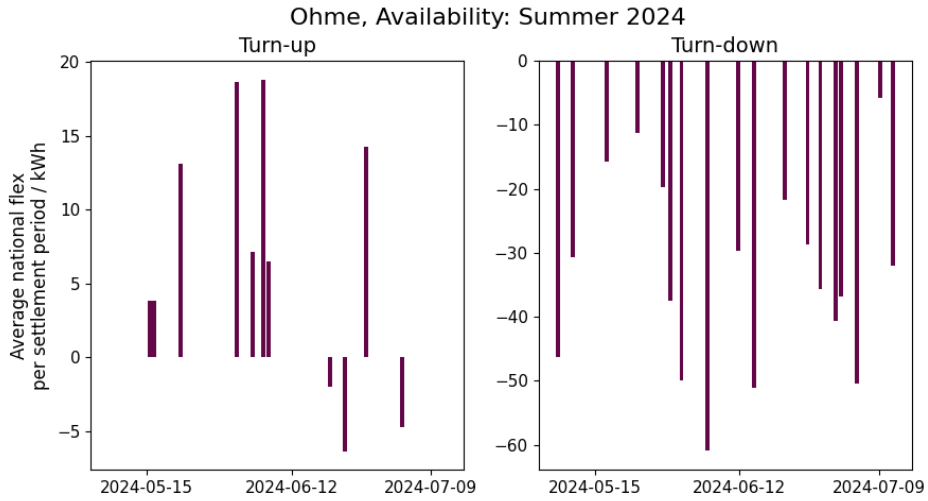

Figure 3: Graphs showing flex delivered over time for each Ohme availability event in summer 2024. Values are averaged over settlement period and summed over GSP. Note that due to the small participant count for these trials, data was only collected for 19 GSPs.

Figure 4 shows the analogous plot for the OVO summer 2024 trial. This was a utilisation trial (involving flexibility of whole household usage) and had more participants, which is the main driver behind the larger flex numbers observed. In both of these trial results, we see high variability in the amount of flexibility that is delivered, but so far this is most extreme in the OVO turn-down case, where the total delivered flex spans three orders of magnitude and includes one 'wrong-way' event.

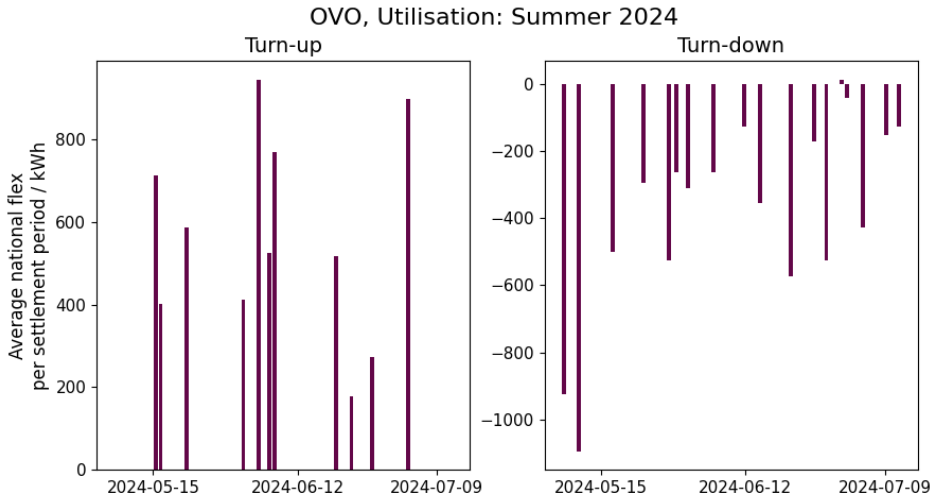

Figure 4: Graphs showing flex delivered over time for each OVO utilisation event in summer 2024. Values are averaged over settlement period and summed over GSP.

Figure 5 and Figure 6 show the amounts of flexibility delivered in the winter 2024 availability trials for Ohme and OVO respectively. These are the largest set of flexibility trials that were carried out. The two sets of plots are broadly similar. The OVO turn-down plot shows a reasonable number of 'wrong way' events. These could be genuine, or down to inaccuracies in baselining. Note that OVO chose to use the control group for baselining here, because the high frequency of events limited the applicability of the P376 baseline, which relies on a recent history of event-free days.

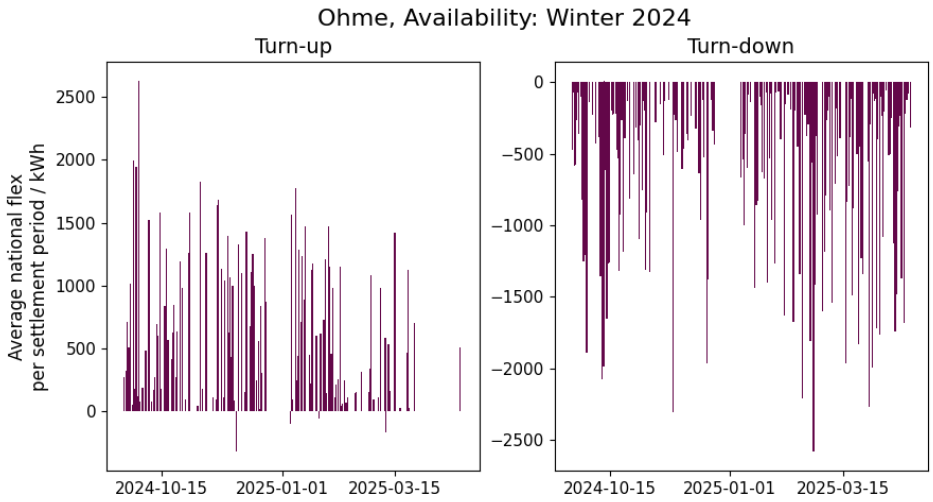

Figure 5: Graphs showing flex delivered over time for each Ohme availability event in winter 2024. Values are averaged over settlement period and summed over GSP.

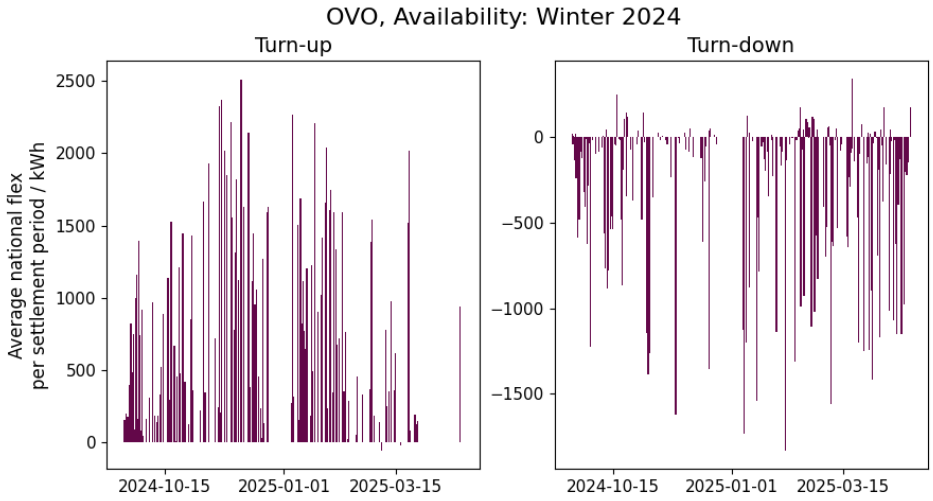

Figure 6: Graphs showing flex delivered over time for each OVO availability event in winter 2024. Values are averaged over settlement period and summed over GSP.

Figure 7 shows the flexibility delivered by each event during the OVO winter 2024 utilisation trials. Here, we see a new pattern – there is a noticeable downwards drift in flexibility over the course of the trials, affecting both turn-up and turn-down. This has two main causes:

1. Issues with baseline accuracy, particularly when temperatures were changing noticeably over the course of weeks (this was a known issue with the P376 baseline during these trials and has been discussed by OVO and others).
2. The presence of eight 'critical down' events, all later in the trials. During these trials, customers were offered larger than usual incentives to change their behaviour during flexibility events.

Unless NESO estimate baselines from raw data themselves, which would require DSRSPs agreeing to share continuous meter data, baseline accuracy will be under the control of each DSRSP. 'Critical down' events and other effects of varying incentive cannot be accounted for within the AFM and EDM so long as price factors cannot be included. We expect issues with baselining, and the potential for DSRSPs to offer customers different incentive levels will persist as two significant factors affecting estimates of flexibility delivery as the models move into BAU.

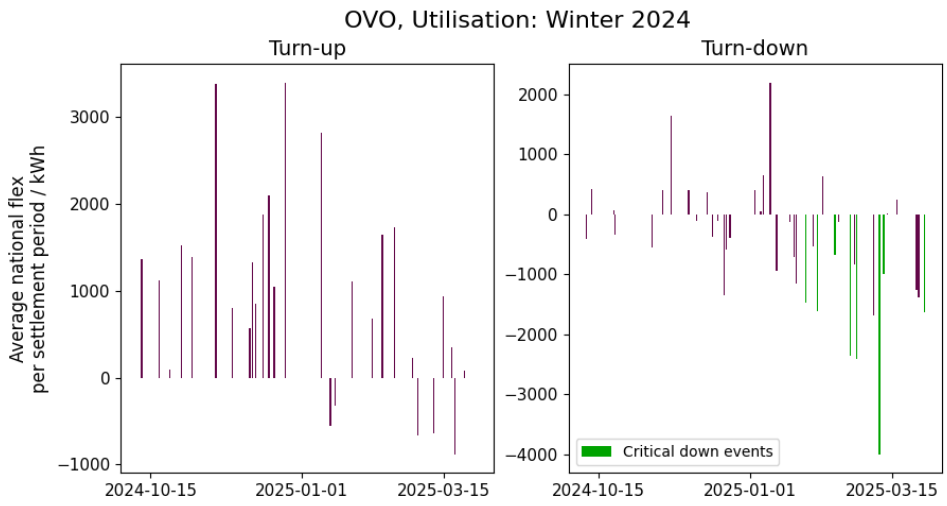

Figure 7: Graphs showing flex delivered over time for each OVO utilisation event in winter 2024. Values are averaged over settlement period and summed over GSP.

Figure 8 and Figure 9 show the flexibility delivered during the summer 2025 availability trials for Ohme and OVO respectively. These have a small number of 'wrong way' events and continue the general behaviour seen in winter 2024. The delivered OVO turn-down flexibility is notably lower than for turn-up or when compared to Ohme turn-down response. This could be due to differing customer demographics or pre-existing optimisation of charging limiting capacity for shifting when charging occurs, among other factors.

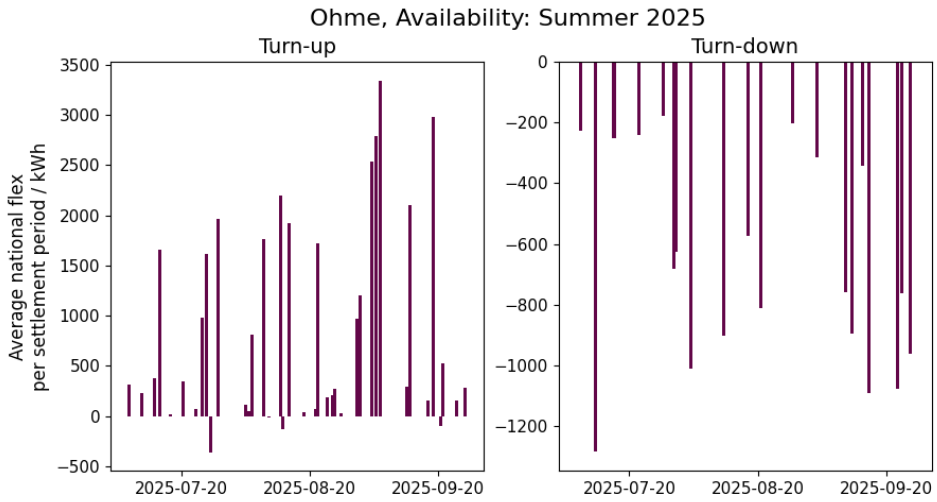

Figure 8: Graphs showing flex delivered over time for each Ohme availability event in summer 2025. Values are averaged over settlement period and summed over GSP.

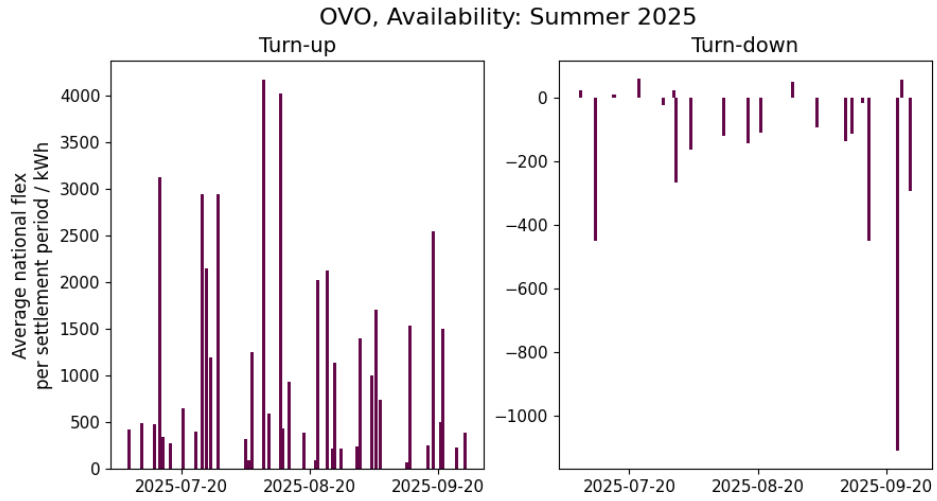

Figure 9: Graphs showing flex delivered over time for each OVO availability event in summer 2025. Values are averaged over settlement period and summed over GSP.

Finally, we show in Figure 10 the flexibility delivered by OVO utilisation events in summer 2025. The turn-up events showed strong responses but the turn-down events did not show any clear shift in demand. These are likely affected by the same baseline issues as previous utilisation trials. This is possibly also affecting the turn-up flexibility, albeit by making the delivered flexibility look stronger

than it actually was. Note that there were only 12 turn-down events and only 18% of customers were asked to participate[10], which could also be a factor in these inconclusive results.

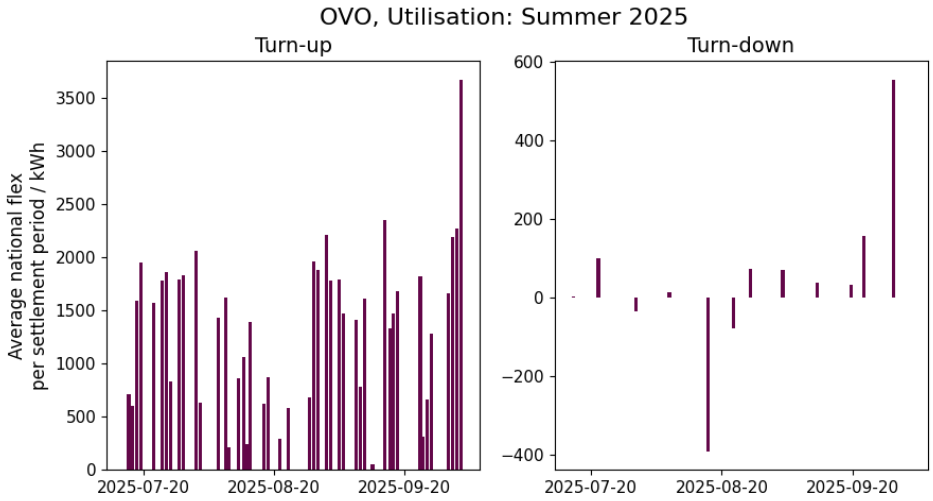

Figure 10: Graphs showing flex delivered over time for each OVO utilisation event in summer 2025. Values are averaged over settlement period and summed over GSP.

We also show in Figure 11 the overall distributions of the flexibility delivered in each settlement period (summed over GSP). These are effectively the distributions that the AFM is aiming to characterise and estimate quantiles for (as other inputs vary). The distributions all peak sharply close to zero, suggesting that it is common to have only small amounts of flexibility delivered. They all have an amount of 'wrong way' response as observed already – suggesting there may always be scenarios where demand shifts in the opposite direction to that requested. Further analysis of the accuracy of baselines is necessary to know how much of this is genuine customer behaviour.

[10] The summer 2025 utilisation trial primarily aimed to test turn-up with several treatment groups to test the response to different incentives, of which only one treatment group also included invitations to turn-down events.

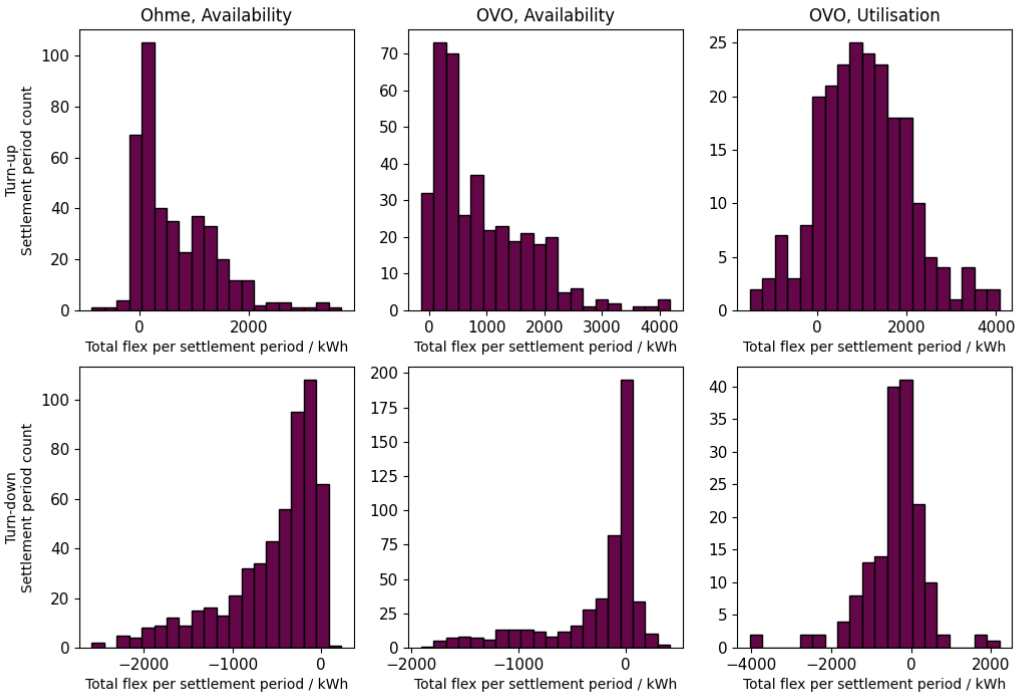

Figure 11: Graphs showing the distribution of flexibility delivered across all settlement periods. Values are summed over GSP to give national estimates. For example, a bar of height 100 between 0kWh and 200 kWh indicate 100 settlement periods saw total flexibility in that range (out of all settlement periods in all events).

The distributions all have long tails in the 'correct' flex direction, indicating that there are also scenarios where response to a flex event is enthusiastic. This is variation that our models capture probabilistically, instead of only providing point forecasts.

Delivered flexibility across the trials

We have already discussed how problems with baselines have likely affected the data being used to train our models. Here, we plot the baseline, actuals, and flexibility over the course of the trial to show seasonal trends. Unlike the plots above, we look at median performance – so the values are representative of what might be expected at the level of a single GSP. We also show the interquartile range (the gap between the 0.75 and 0.25 quantiles) in the form of error bars. Note that the quantiles for delivered flex are calculated after subtracting the baseline from the actual usage – the values will therefore not be equal to the difference between the median baseline and median actuals.

We did not have access to timely participant data throughout the course of the trials so have not attempted to correct for variations in participant number across the trials. This mostly affects the early data for Ohme, where the number of participants in the summer 2024 availability trial was much smaller than in subsequent trials.

We notice throughout Figure 12, Figure 13, and Figure 14 that the error bars (the gap between the 0.25 and 0.75 quantiles) are very large compared to the median. This suggests that it may be challenging to give estimates of expected flexibility that are both useful and reliable. For example, the value for the amount of flexibility that is achievable in 95% of turn-up events may well be less than zero. Furthermore, the median delivered flex is frequently close to zero, indicating that the delivered flexibility observed in Summaries of data is often dominated by a smaller number of GSPs, which may change between events. The method used to produce these plots is also different to analysis of the flexibility delivered elsewhere – it is not suitable for giving firm estimates of the efficacy of flex events.

Figure 12 shows the seasonal plots for the Ohme availability trials. The first three months here are where the participant count was comparatively low, which means the levels of delivered flexibility will be much more subject to randomness from individual consumer behaviour. There appear to be lower amounts of turn-down flexibility delivered in summer 2025, with the actual usage during the events much higher than previously. This could be down to the scheduling of these trials which specifically targeted peak and off-peak times of day (shown below in Figure 17). The scheduling also explains the differences in baseline energy usage between the turn-up and turn-down events in summer 2025 (not seen in earlier trials). This trend is also seen in the OVO data.

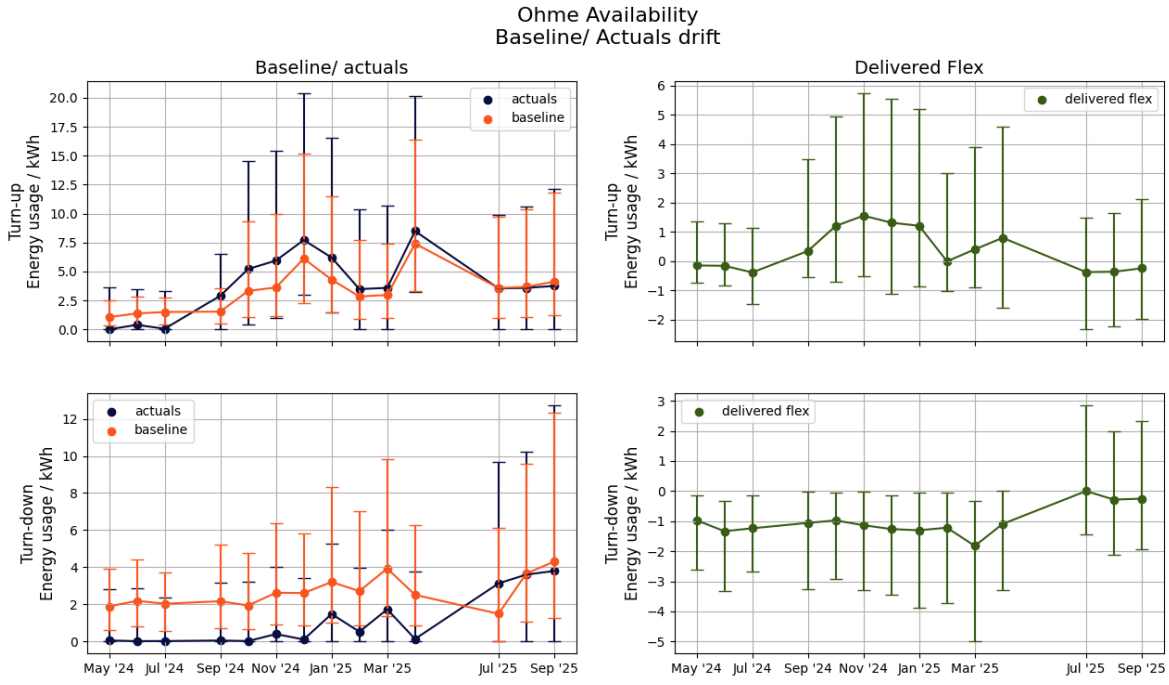

Figure 12: Plots showing seasonal variation in baseline and actual energy usage over the course of the CrowdFlex trials for Ohme availability, along with the delivered flex.

In Figure 13, we show the seasonal variation for the OVO availability trials. Note that these only started in winter 2024. We see that in the winter trials especially, the error bar in the direction of the requested flexibility is much larger. This is partly caused by variation in participant count

across GSP. The largest GSPs are capable of delivering the most flexibility, and our data for them is less subject to random variation.

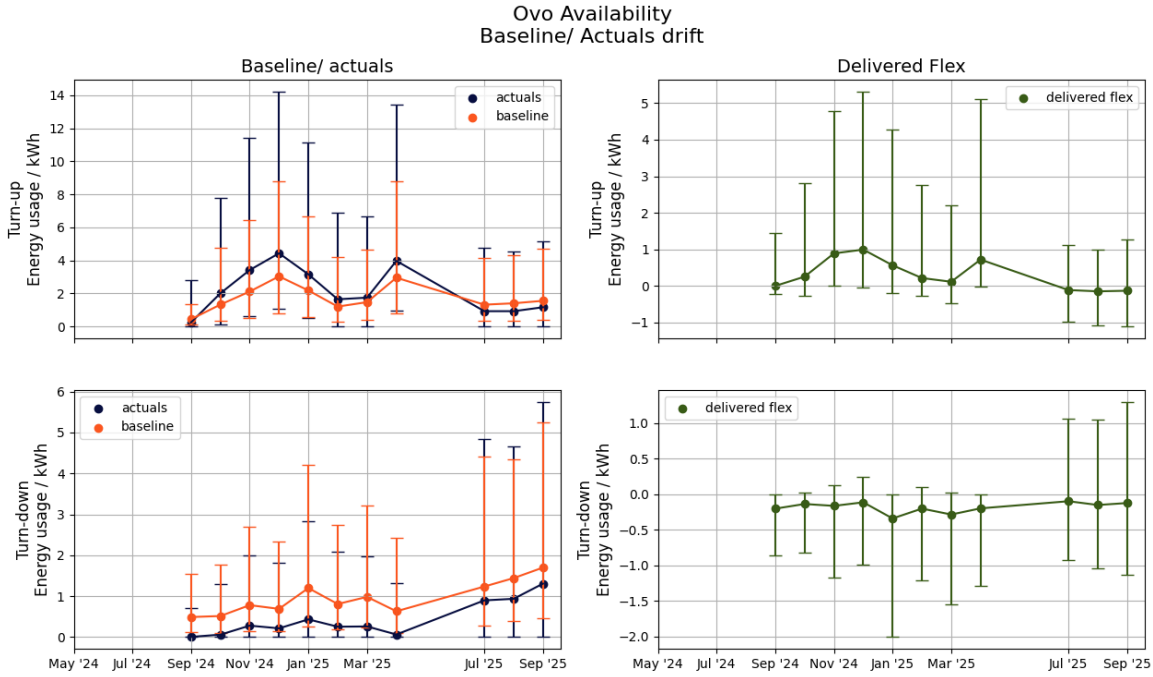

Figure 13: Plots showing seasonal variation in baseline and actual energy usage over the course of the CrowdFlex trials for OVO availability, along with the delivered flex.

Figure 14 shows the variation across time for the utilisation trials. These were the trials where there were especially notable baseline issues (particularly for turn-down). We see here that the median baseline and actuals tend to be very close together. Despite the apparently strong performance in Figure 10 of the summer 2025 turn-up events, this representation suggests that it was not always conclusive at the level of individual GSPs. It may be that the total delivered flex is dominated by a small number of large or well-performing GSPs. In order to examine this further, we ideally would need access to accurate participant counts for each event.

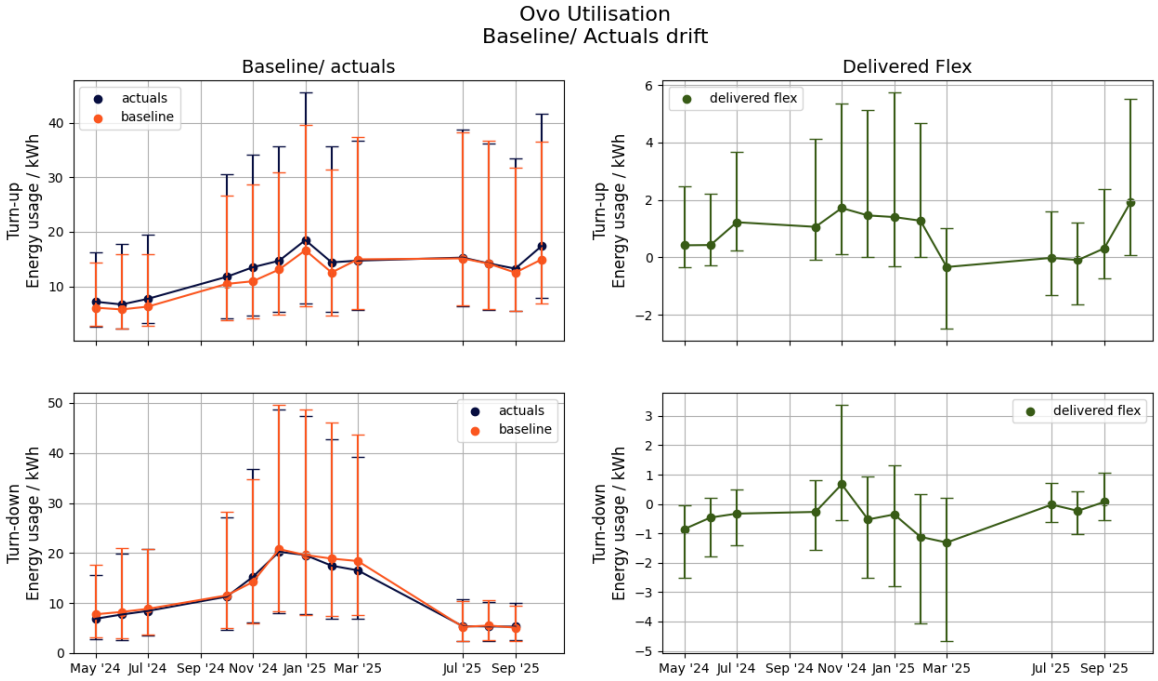

Figure 14: Plots showing seasonal variation in baseline and actual energy usage over the course of the CrowdFlex trials for OVO utilisation, along with the delivered flex.

Data domain and applicability to BAU

An important consideration from the CrowdFlex trials is the extent to which the data is useful into BAU. There are ways in which fixed flexibility trials will always be unrepresentative of general use.

- Frequency and scheduling of events:
  During parts of the flexibility trials, there were multiple events per day (availability) or per week (utilisation). In both cases, this pace risks 'using up' available flexibility, which would need time to replenish. For availability, an EV might already be fully charged. For utilisation, the consumer may either be unable to flex their usage any further or reduce their response due to messaging fatigue. If, in BAU, flex events are separated enough as to be entirely independent, these effects may be less prevalent (but new data will be needed to explore this and provide new training data). It's also possible that DSRSPs might decide to use different customers in different flexibility events to mitigate these issues.
- Flexibility procurement:
  Trials were scheduled in advance and with specific consumer incentives agreed many weeks before the market conditions at the time of the event are known. Under more 'normal' market conditions, the price that NESO would pay DSRSPs and other markets that DSRSPs are participating in are likely to vary. Both factors may affect when and how much DSRSPs can instruct EVs to charge (like in availability trials). They can also influence the incentives offered to consumers to plug in EVs (availability) or flex their usage (utilisation), and thus their collective response.

- DSRSP characteristics:
  Data was collected only for participating Ohme and OVO customers (and OVO-only for utilisation). It is unknown if they (and their customer bases) will be generally representative of other DSRSPs.
- Data granularity:
  Data was collected at the level of GSP and settlement period and this is the level the models generally predict at. It is possible (with appropriate care) to use the same data to predict at a coarser level (at the level of hours or for specific groups of GSPs) so long as this is known about when building and training models. Indeed, considering geographic regions larger than single GSPs may reduce the variability of flex forecasts as the randomness inherent to small participant counts will no longer be as prevalent. It will not however generally be possible to predict at a finer granularity, except by assuming that the current data is distributed evenly across time or in proportion to household across each GSP.
- Participant demographics:
  The number of households in the trials was limited to predefined quotas, and to consumers who opted in to receiving notifications. We do not know if they will be representative of the general population when flexibility participation is rolled out more widely, including to customers who may be less engaged with communications received from their energy provider.
- Baseline methodology and drift:
  Baseline accuracy has been an ongoing concern. Adjustments to how this is calculated will affect the validity of the training data that has already been collected. Similarly, the event frequency difference mentioned above means that P376 baselines in BAU may be more accurate. This is beneficial in the long term, but the data collected from the CrowdFlex trials may become less useful for making BAU predictions. If this occurs, it will introduce a source of systematic error that is best dealt with by collecting data from new flex events to reduce the importance of older trial data in model training whilst carefully monitoring model performance on each dataset.
- Target delivery volume:
  During trials, DSRSPs aimed to deliver as much flexibility as possible given the agreed incentives and nudges. However, in BAU, taking the Demand Flexibility Service (DFS) as an example, we expect DSRSPs to be dispatched to deliver a particular amount of flexibility, which may be less than the maximum flexibility available.
- Trial event characteristics:
  Input data seen by the flexibility models may not be representative of the full range of behaviour that will be modelled under BAU. For example, where limitations in trial design restricted what was possible. This is something that can cut both ways – if all BAU events are expected to follow a particular pattern (for example, of short duration) then data collected outside this will be of limited value. However, if the models are required to be flexible or there is uncertainty as to what BAU flexibility events may look like, then data covering the full range of expected inputs is vital to ensure the model outputs will be valid. Of particular difficulty is

that if no training data for a particular event type has been collected, then it is extremely hard to estimate model performance on that event type. Some model types (especially linear ones) also benefit from having a broad range of inputs in the training data, even when the range they need to predict on in practice is more restricted.
There is a separate work package within CrowdFlex to explore in more detail how the trials data should be used for BAU models and to give recommendations of how best to maximise the value of future data collection.

We now analyse the characteristics of the flexibility trial events in more detail, breaking down the variation seen in the corresponding model inputs.

Figure 15 and Figure 16 show the relative frequency of different event durations in the collected data. For availability, we see a good range of event durations between 0.5-2 hours, especially in the EDM data collected for summer 2025. There are few three-hour events (and none in the set with event shoulders) so predictions for events longer than two hours are unlikely to be reliable. For utilisation (Figure 16), one-hour events dominate (and is the only duration where shoulder data was collected). Predictions (especially for the EDM) are unlikely to show meaningful variation as event duration varies. In general, we expect that delivered flexibility may not vary a lot across the course of a flex event, but there will be some threshold beyond which additional flex is not available. We do not currently know what event length this corresponds to (and it will likely vary based on trial type, time of day, and day of the week).

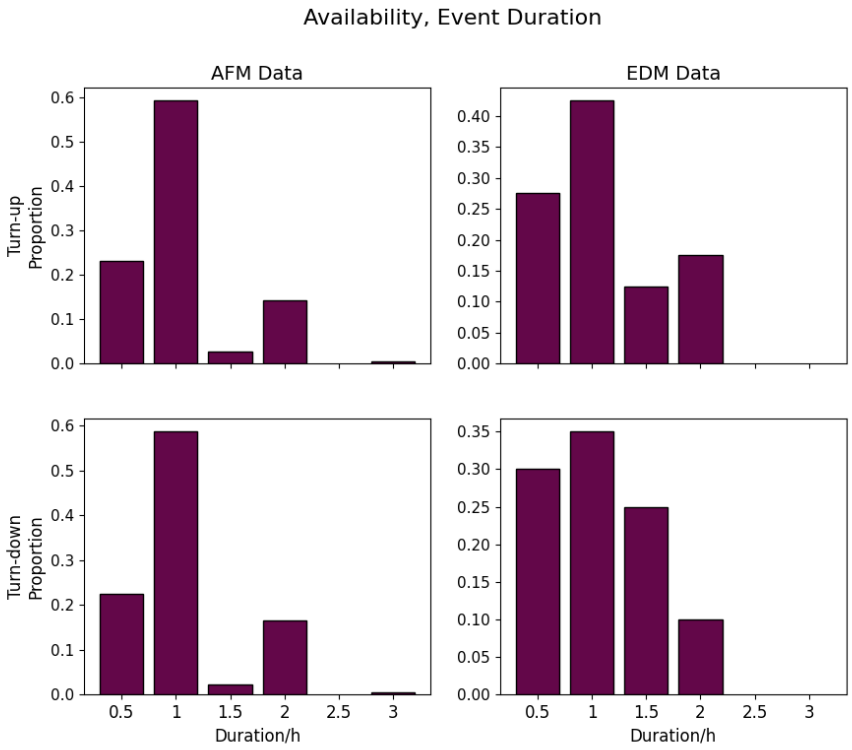

Figure 15: Plots showing the proportion of different event durations in the collected data for availability trials. AFM data contains all flexibility events, EDM data restricts to summer 2025 (where data was also collected for the event shoulders). Proportions are counted by settlement period, not by event.

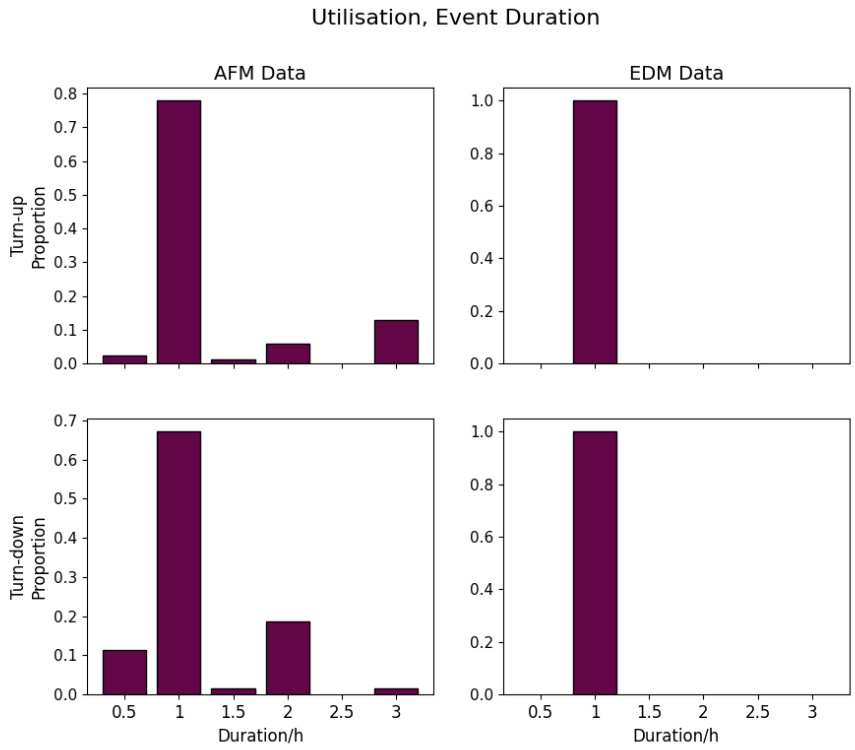

Figure 16: Plots showing the proportion of different event durations in the collected data for utilisation trials. AFM data contains all flexibility events, EDM data restricts to summer 2025 (where data was also collected for the event shoulders). Proportions are counted by settlement period, not by event.

In Figure 17, we show the time of day at which availability flex events started. We see that the AFM availability data has at least some representation at all times, although the relative freqency of certain times varies between turn-up and turn-down. However, the summer 2025 trials restricted turn-up and turn-down events to only particular times of day (which were mutually exclusive). It will therefore not be possible to provide EDM predictions for all times of day using the currently available training data.

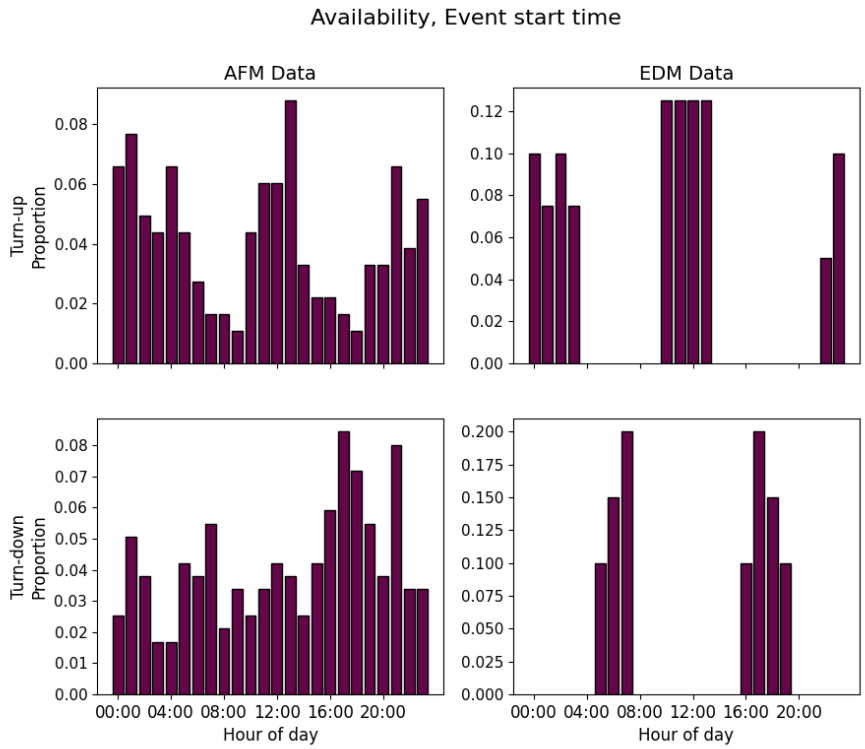

Figure 17: Plots showing the proportion of different event times in the collected data for utilisation trials. AFM data contains all flexibility events, EDM data restricts to summer 2025 (where data was also collected for the event shoulders). Proportions are counted by settlement period, not by event.

Figure 18 shows the equivalent plots for utilisation. Here, the start times are more restricted. Some of this is to be expected – flexibility relying on active changes in behaviour will generally show poorer performance if it occurs during times when homes are empty or when participants are asleep[11]. Data for such flex events is therefore of less value or interest. We nevertheless have the same problem as for availability – if the assumptions around which times of day are most likely to need flex events turn out to be incorrect, then the models (EDM in particular) will be unable to generalise without more training data. Time of day is believed to be an important factor in how much flex is available so extrapolation here is not advisable.

[11] Although usage of smart appliances or timed cycles on (for example) washing machines can still provide some level of flexibility.

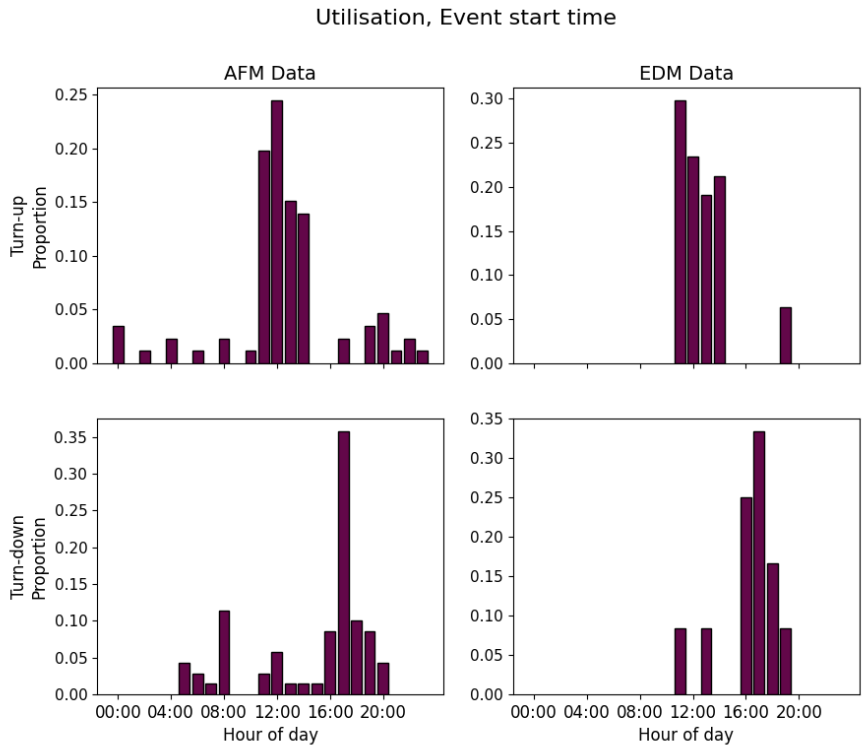

Figure 18: Plots showing the proportion of different event times in the collected data for utilisation trials. AFM data contains all flexibility events, EDM data restricts to summer 2025 (where data was also collected for the event shoulders).

We finally consider day of the week. Figure 19 shows the relative frequencies of each weekday (for events which split across more than one calendar day, it is the day when the event starts that we consider) for availability events. Across all the data (AFM case) there is good coverage of all weekdays. For the summer 2025 data (EDM, with shoulder) there is fair coverage for turn-up events but very few turn-down events on Friday and none on Saturday. This could be alleviated by having the model take weekday/weekend as an input instead of the exact day, but more analysis should be carried out to establish if this is sufficiently informative. Behaviour on a Friday is often distinct to that on other weekdays and there are often also differences between Saturdays and Sundays. It may be necessary to establish if these differences are important when considering energy flexibility and usage or charging of EVs.

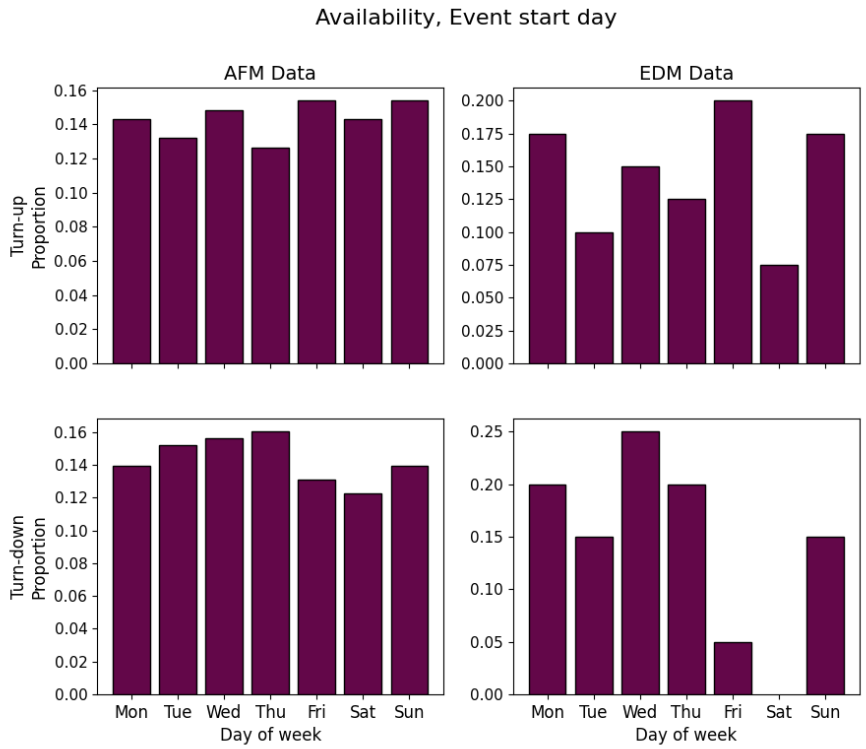

Figure 19: Plots showing the proportion of different event days in the collected data for availability trials. AFM data contains all flexibility events, EDM data restricts to summer 2025 (where data was also collected for the event shoulders).

Finally in Figure 20 we show the relative frequency of different weekdays for utilisation. For both AFM and EDM and across both flex directions, there is reasonable coverage of all weekdays but less data for weekends (especially for turn-down and EDM). Again, modelling this model input as weekday versus weekend may be sufficient but more weekend data (especially for Sunday) is likely needed to establish if the data that is collected already is actually representative of all types of behaviour. We could have taken a more rigorous, quantitative approach using chi-squared or another statistical test to compare distributions of training and validation data, however since the intended application is to model BAU flexibility events, this would be of limited value for long-term planning at this stage.

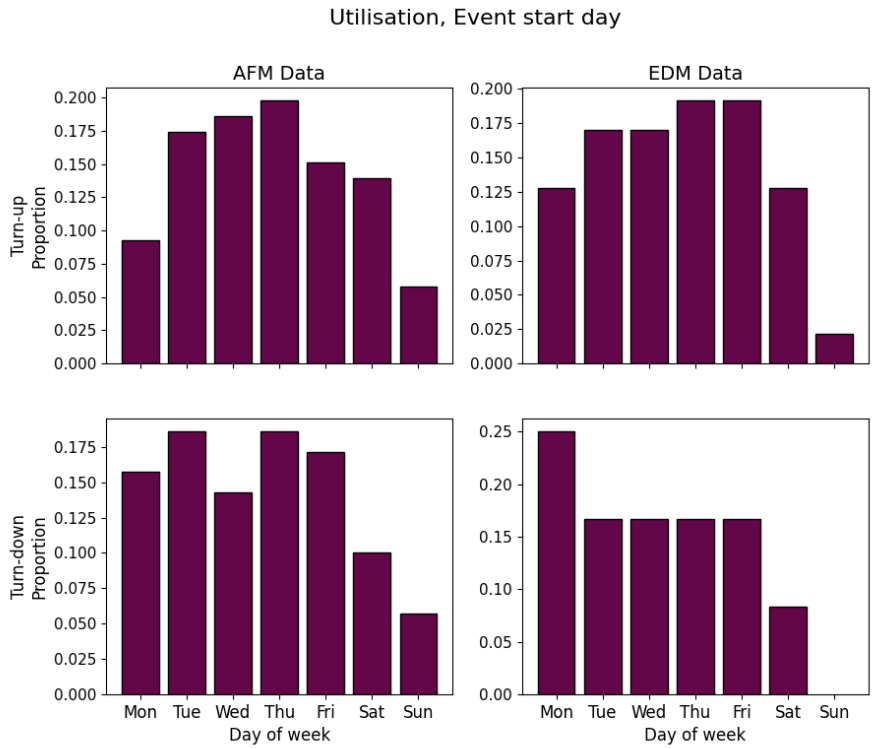

Figure 20: Plots showing the proportion of different event days in the collected data for utilisation trials. AFM data contains all flexibility events, EDM data restricts to summer 2025 (where data was also collected for the event shoulders).

3. Modelling

3.1 Model selection

We trialled two types of model for the AFM and EDM: linear quantile regression (LQR) and gradient-boosted trees (GBT). In a LQR model, the output is a linear combination of features, and each quantile is represented by a different model. A GBT model uses a group of decision trees to predict the output, where a decision tree is a graphical representation of a decision-making process in which nodes represent decisions and leaves the output. The GBT algorithm fits trees sequentially so that later trees improve upon errors in earlier trees.

While LQR models are more easily explained and understood, we found that GBT models provided better performance, both in terms of accuracy and fitting time. A LightGBM implementation of a GBT model proved the most accurate and computationally tractable, and we therefore recommend this model architecture and have used LightGBM for the final model deployment and evaluation.

3.2 Model architecture and features

Model specifications

Report D1-1 model specification describes the models as originally intended. However, we agreed several changes with the DSRSPs during the development cycle. The wording here supersedes D1-1. Further documentation is in the 'CFLX' code repository with a summary of inputs in Appendix B: Model input data.

AFM

We assume that during BAU, the AFM will use the following inputs to forecast the total available flexibility across DSRSPs:

- **Demand forecasts** from each DSRSP describing how much energy they expect their dispatchable consumers/assets to use if no domestic flexibility is dispatched in any market. These point forecasts should be provided for each grid supply point (GSP) for which the DSRSP is willing to provide domestic flexibility. These demand forecasts play a similar role to the Physical Notifications (PNs) of units within the Balancing Mechanism, hence should only be reported for those customers who may be incentivised to change their demand. These demand forecasts will be collected from the DSRSPs via an established API, specified in D1-2.
- **Weather forecasts**. These forecasts should be sufficiently localised that they apply separately to individual GSPs. See Feature engineering for the set of weather features we selected after experimentation during model development.
- **Time of event** including the hour, day of the week and month.
- Details of **special events** such as bank holidays, key cultural or sporting events etc.
- **Direction** of the flexibility required — either 'turn-up' or 'turn-down'.
- **Region** to predict – see Model regions for options.
- **Lead time** (the difference between forecast generation time and target time) for both demand and weather forecasts.
- **Duration** of the event, in hours.

Using these inputs, the AFM will return the quantiles of the forecasted distribution of the total available flexibility.

The model assumes that NESO are wanting to procure continual flexibility over the whole window. Thus, it will attempt to model how the available flexibility may diminish over the duration of the window. For instance, charging EVs can only turn-up their demand until they are fully charged, thus limiting the window over which flexibility is available.

EDM

The EDM uses the same set of inputs as the AFM to forecast the expected delivery from a combination of bids/offers from multiple DSRSPs, except:

- **Demand forecasts** now describe how much energy they expect their dispatchable consumers/assets to use if this flexibility event does not go ahead. Unlike the AFM where the demand forecasts assume no flexibility is dispatched in any market, demand forecasts for the EDM still need to account for any other dispatches, including into other flexibility markets. Thus, forecasts for the AFM and EDM will differ for a given time point if the DSRSP expects to also dispatch the consumer/asset into a different flexibility market. For summer 2025 trials, DSRSPs avoided dispatching into multiple markets simultaneously. Thus, the demand forecasts for the AFM and EDM are identical, and we could collect them through the same API.
- We require demand forecasts (and actuals for training) for the continuous window from six hours before until six hours after the proposed event. This enables prediction of demand shift, such as possible turn-down prior to a turn-up event. Although demand may shift beyond six hours either side of an event, the consortium agreed this horizon as a balance between capturing demand shift and avoiding interactions between events; wider shoulders would have made it very difficult to schedule adjacent CrowdFlex trial events such that their shoulders do not overlap.

And it takes the following additional inputs:

- **Target flexibility** indicating the total amount of flexibility the DSRSP(s) are aiming to deliver, given by the amounts on their bids/offers. For the 'GSP' region, this is the target flex per GSP. For 'DFS' and 'LCM', this is the total across the whole region.
- **DSRSPs** contributing to the target flexibility.

When querying the EDM, the user inputs the start time and duration for when flexibility is required; the model returns predictions for every settlement period between six hours before and six hours after the requested event. We assume that each event affects delivery only within these six-hour shoulders.

Note that the DSRSPs only populated shoulder data for the summer 2025 trials. We include earlier trials in the training data for the deployed EDM, to maximise the training dataset size and hence quality of fit. However, for evaluation, we train separate EDMs excluding the earlier trials to avoid biasing the evaluation metrics towards in-event times, thereby ensuring a fairer evaluation and one that is more applicable towards BAU when we expect all further training data to include the shoulders.

Both models

Both the AFM and EDM currently predict over five quantiles, (5%, 25%, 50%, 75%, 95%).

In order to convey the variation over time, the models forecast the distribution of flexibility at half-hourly intervals. If required by NESO, these can be mapped to cardinal points during post-processing.

Internally, the models predict in kWh per half-hour period, although user interfaces may choose to convert these to other units for display. Demand forecasts, target flex and actual flexibility are all in these units.

The models are agnostic to price. The relationship between the price NESO would see (and pay) and delivered flexibility is complex and may not be stable enough for NESO to use in decision making. In particular, there is a disconnect between the price NESO would pay and the incentives offered to consumers, which is a commercial decision taken by each DSRSP. Moreover, introducing a public notion of price into the models opens the possibility of market manipulation, further reducing stability of the models. The CrowdFlex trials have the further issue of the incentives offered being independent of the current state of the market, so there is no meaningful public price for the trials anyway. Therefore, we have excluded any concept of price from the models.

Note that DSRSPs could have extended their APIs to include their forecast availability, based on their own models of their consumers and assets. However, the forecast availability provided by DSRSPs would likely be closely linked to the price they will ask for to provide that volume. For the same reasons as excluding price from the models, we do not include availability forecasts from DSRSPs.

Feature engineering

From the inputs in the model specifications, the chosen AFM features are:

- GSP ID,
- Forecast demand,
- Forecast lead time,
- Month,
- Day of the week,
- Hour,
- Time since the start of the event,
- Weather lead time,
- Rainfall,
- Humidity,
- Central estimate of temperature,
- Event duration.

The EDM extends this list to include:

- Target flex,
- Time through shoulder (time until start or time after end of event).

The model features have been selected by performing model evaluations throughout the project. They include a mixture of original inputs and derived quantities that lead to good model performance.

We fit a model for each combination of flex direction, region (see Model regions) and quantile. Of the model features, not all came directly from the data: we construct temperature as the mean of the minimum and maximum temperatures in each forecast period, the temporal features such as day of the week and month are extracted from the forecast target time, and we derive the 'time since start' and 'time through shoulder' columns from the forecast target time and duration.

Target flex is the total amount of flex the user is aiming to achieve per settlement period. In BAU, we expect this to be the total flex procured. However, in the trials, DSRSPs were not aiming to deliver a particular volume of flex, rather attempting to generate the maximum flex possible. Therefore, target flex is not available for training the EDM and a proxy is required: we used AFM predictions of the 50% quantile as a value representative of likely delivery[12]. We do not expect this to accurately reflect the target flex in BAU, and once enough data has been collected, this training data should be discarded.

Furthermore, this effect may reverse in BAU: DSRSPs may only deliver flexibility for which they were dispatched, which may be less than the maximum flexibility available across their customers. This would lead to underreporting in the BAU training data for the AFM. Further work is required to plan training data collection for the AFM and EDM in BAU to mitigate this. Alternatively, it may be possible to adapt the AFM to use data from events where a specific value of flexibility was targeted.

GSP ID and temporal variables are target-encoded. A different style of encoding, such as periodic encoding, could be more suitable for temporal variables. Numerical features are scaled by subtracting the mean and dividing by the variance. This ensures all features are of comparable size.

For the AFM, we train one model per DSRSP and trial type. The deployed model is the one trained on Ohme availability data[13]. Enabling other AFMs would require the creation of additional endpoints (see Infrastructure). The EDM, on the other hand, has one model per trial type but with the same model covering all DSRSPs, so that it can predict for either DSRSP individually, or for the flexibility that can be provided by both of their customer bases simultaneously.

---

[12] We created an extra training data table for proxy target flex to sit alongside those created by the gold pipelines. We manually run a Python script to update this table as necessary before any manual refitting of the EDM. Maintaining a separate table ensures the proxy can be easily removed for BAU.

[13] Where Ohme inference data is unavailable, the model uses OVO inference data instead; the user should acknowledge that the predictions will be of inferior quality in this situation.

The deployed EDM is the one trained on availability data, matching the AFM[14]. We assume all DSRSPs behave equivalently and exclude DSRSP ID as a feature of the model. This is necessary because each DSRSP may deliver varying proportions of the total flex, requiring a more complex feature than just which DSRSPs are included. Furthermore, we aggregate over DSRSPs in the training data to create additional data points corresponding to when more than one DSRSP should be included in the predictions: we take the mean of weather variables and sum the forecast, actual and predicted demands. In all cases, we avoid aggregating turn-up data together with turn-down data. We handle inference in the same way. The EDM does not predict for times and GSPs where not all requested DSRSPs have provided demand forecasts. It is worth noting that this aggregation changes the information contained in the target-encoding of GSPs. It may be that switching the order in which aggregation and encoding occur improves the accuracy of predictions.

Model regions

Typically, the user requires predictions for multiple GSPs. A bespoke model for each region of GSPs is most accurate. One could instead naïvely predict for each GSP independently and then sum quantiles, but this increases the variance, so could require control room engineers to take more expensive actions to compensate. This pushing of extreme quantiles further from the median occurs because summing over quantiles assumes that when one GSP delivers at a given quantile, all deliver at that quantile. Thus, all GSPs deliver an extreme response at once when in practice one extreme response from a GSP is likely to be counteracted by other GSPs nearer their medians. Furthermore, the median may not be correct if the data is skewed, which we have observed it to be.

Therefore, the AFM and EDM give the user a choice of three 'model regions':

1. 'DFS' – we model total predicted flex across all GSPs, the same as those in the Demand Flexibility Service.
2. 'LCM' – we model total predicted flex across all GSPs in the Local Constraint Market.
3. 'GSP' – we model predicted flex for each GSP individually then sum quantiles naïvely.

[14] Note that when using AFM predictions as a proxy for target flex, the AFM predictions remain for Ohme availability, regardless of whether the EDM is being trained on utilisation data and whether the training data point is in fact for OVO. We do not expect this disparity to affect the EDM any more significantly than using a proxy in the first place. This observation will become redundant in BAU.

For the models trained on a whole region ('DFS' or 'LCM'), we sum or average over variables in the training and inference data in the same manner as for aggregating over DSRSPs in the EDM[15]. Other regions may be added, but they will need to be configured, and the corresponding model trained before use. Adding or removing a GSP requires defining a new region.

The region covered by the LCM varies between days, according to the needs of the grid. It consists of Scottish GSPs. NESO requested that the 'LCM' region for the CrowdFlex model should include all GSPs that participated in LCM at any point during 2024. The region for each date is defined in a GeoJSON file available on Piclo[16]. We then identified the GSPs in the 'LCM' region as those whose centroid is inside any region in the GeoJSON file. A visual inspection of the map confirmed that no non-convex GSPs had been omitted due to their centroid lying outside their area.

Despite challenges with combining quantiles across GSPs, we also model the 'GSP' region, because it is the most flexible approach. The user can filter which GSPs to include by passing a list to the model endpoint. However, we did not implement this functionality in the user interface because of resulting high variance and NESO requesting we focus on the 'DFS' and 'LCM' regions.

Hyperparameter tuning

The libraries we used for gradient-boosted trees have three parameters which we tuned (see Appendix C: Model implementation details for details) to improve the accuracy of the models:

- The learning rate or step size shrinkage, which determines how much each new tree contributes to the model. Large values of learning rate may cause overfitting which means that the model performs well on the training set, but poorly on new data. In contrast, low values slow down the fitting of the model so that more trees are required.
- Maximum depth, which controls the number of layers in each tree.
- Number of estimators, the number of trees to be fitted in the model.

Overly small or large selections for maximum depth and number of estimators can produce underfit or overfit models respectively.

Infrastructure

As part of the project, we selected appropriate services to hold the data and models as well as all the required processing and connectivity. We host both models within Azure Machine Learning

[15] We drop anti-events from the training data for these regions because it does not make sense for a region to split its delivery direction. Moreover, a GSP turning up counteracting one turning down would result in little net flex across the region.

[16] https://piclo-open-data.s3.eu-west-1.amazonaws.com/public-data/ngeso_lcm_competition_boundaries_2024.geojson

(Azure ML). It is a component of the Microsoft Azure ecosystem that is well-suited to hosting and monitoring such models. It sits alongside Azure Data Factory for data processing and Azure blob storage for the saved data (see in 2.2 Data preprocessing and integration and Appendix A: Data pipelines implementation details). The code for data processing, the models and the user interface (see 3.5 User interface), and accompanying documentation are in the 'CFLX' repository in Azure DevOps[17]. Keeping our entire system within Azure facilitates security and integration between the various components.

There is an API endpoint for each model. The user interacts with the models by sending requests through the UI, which queries the endpoints. Other software, such as tools within NESO's control room, can obtain predictions from the models by querying the same endpoints

The caller encodes their input parameters as a JSON in the body of a POST request to the API. The specification and an example are in the model endpoints section of the 'CFLX' code documentation. The AFM takes:

- Start time,
- Duration,
- Flex direction,
- Region,
- List of GSPs to filter to (optional) – only used with 'GSP' region and cannot be specified via UI,
- Username – account submitting request logged as security metadata, not for prediction,
- Time of request –logged as security metadata, not for prediction.

In addition, the EDM takes:

- List of DSRSPs to include,
- Total target flex per settlement period (per GSP in 'GSP' region); may be set to null (blank in UI) to query AFM as proxy for target flex.

On success, the model endpoint returns a JSON file containing the total demand forecast and predicted flex for each settlement period. For the 'GSP' region, it also includes the demand forecast and predicted flex for each GSP (for each settlement period).

The UI plots the graphs from this data. Alternatively, the user can download the JSON file directly by clicking on the 'Export' button for loading into other software. An automated tool is likely to ingest the JSON file directly without requiring a human to click through the UI.

[17] Two further DevOps repositories ('infrastructure' and 'weather') are necessary to complete our build. Their purposes are described in the documentation in the 'CFLX' repository.

Code structure

We packaged our Python code into a modular class structure. Modularity aids maintainability and extensibility to new model types or other features in future. In particular, configuration constants are defined in a single location, which makes it easy to adjust the features selected to include in the model, for example.

Azure ML is built upon MLflow, so we built our classes to be compatible with MLflow. An added advantage is that MLflow provides a framework for automatic tracking of model performance, including between versions of the model. The user can identify when retraining is insufficient to prevent model drift and thus hyperparameter retuning and/or further feature engineering are required.

3.3 Training and validation strategy

To perform model tuning and feature selection we hold back 20% of the trial data as a test set. Candidate models are then trained on the remaining 80% before observing the performance of the resulting model on the held back test set. For previous evaluations, this used the most recent 20% of flex events of each type (turn-up, turn-down, antisymmetric). For the final evaluation, we instead chose each 5th turn-up and turn-down event[18].

We judge the strength of each candidate model based on three main accuracy measures:

- Coverage rate – what proportion of predictions fall below the given quantile.
- Quantile loss - a measurement of the distance between the predicted quantile and the true value. It is analogous to mean absolute error (and identical for the median prediction) but with each error weighted unequally based on the predicted quantile. As an absolute measurement, it must be interpreted relative to the values in the underlying data.
- Scaled quantile loss (SQL) – quantile loss scaled by the quantile loss for a naïve forecast, calculated as the corresponding quantile from the training data in the selected flex direction, and optionally filtered by in event/shoulder (EDM only), with all other input parameters

[18] There were no antisymmetric events in the summer 2025 trials and they are distributed irregularly throughout the earlier trials. Since we only give figures for turn-up/turn-down performance (anti events are split into each category at the GSP level), we opted to simplify the process and include all these events as training data. This doesn't affect regional models, where antisymmetric data cannot be used for training.

Using the previous train/test split method would have meant that the 20% of test data overlapped almost exactly with the summer 2025 data. Instead, since we know there were distinct trends seen in each season, we wanted the final evaluation to represent a broad cross-section of the collected data.

ignored[19]. Values below one indicate the candidate model performs better than the naïve forecast.

Coverage rate is the easiest to interpret and gives the most direct measure of 'how often will my forecasts be above or below the true value?'. A model for the 0.25 quantile should achieve a coverage rate close to 0.25. (Scaled) quantile loss provides good information about the average accuracy of the model. We recommend that both types of measure are considered when evaluating model performance. The evaluation process is shown in Figure 21 and was used to calculate the metrics analysed in 3.4 Model performance.

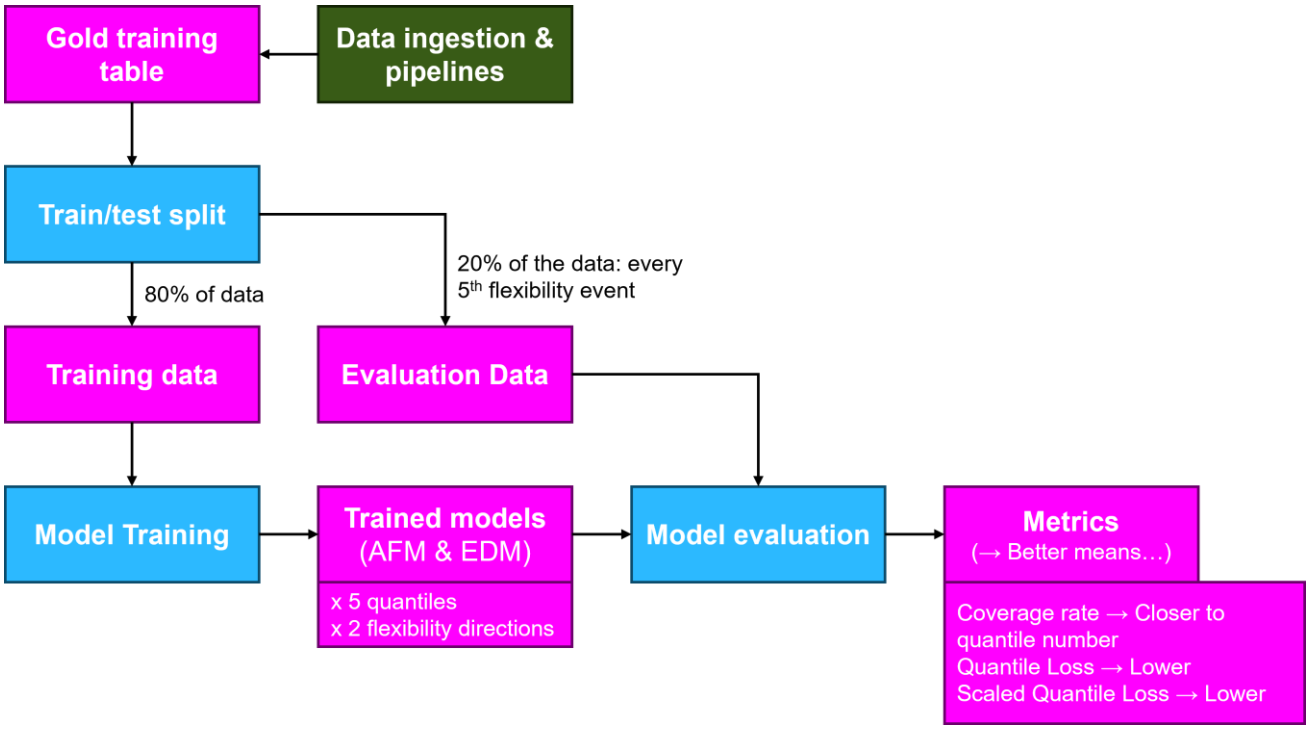

Figure 21: A diagram showing the workflow for evaluating the AFM and EDM using withheld data.

[19] 'Naïve forecast' has a particular meaning when calculating MASE (mean absolute scaled error) of time series forecasts, which scaled quantile loss is adapted from. There, the most recent observation is 'naïvely' used. This is not suitable here as the actuals data is not collected continuously and is not a reasonable estimate for quantiles.

The naïve forecast is not intended to be interpreted as a benchmark against an alternative method – if such methods are later developed then SQL can be used to provide an objective comparison of both.

Model monitoring and maintenance

In order to facilitate ongoing model development, model monitoring and maintenance can be automated to run at regular intervals. We recommend scheduling pipelines with the following components:

- Monitoring: models are evaluated on their performance on recent events as new data is collected.
- Refitting: models are refitted to maximise use of all collected training data.
- Testing: a new model is deployed and tested in a dedicated environment to catch edge cases and bugs that might interrupt service to the live UI.
- Deployment: deploy the new model to the endpoint to be used by the UI.

These outputs from monitoring should be tracked over time and include both specific measurements on recent trials ('how well did we forecast flexibility just now?') and metrics calculated on broad evaluation sets ('how are model metrics changing over time?'). Tracking these scores (quantile loss, scaled quantile loss, and coverage) will allow users to be confident in the quality of predictions being displayed on the UI.

Changes in scores may indicate model drift, which is expected when modelling complex behaviour over months and years. This then allows for targeted intervention by a modelling team to maintain model accuracy. For example, we recommend discarding the trial data when sufficient BAU data has been collected, as the trials operated under different circumstances and so their data will be less representative of later domestic flexibility. The optimal features and hyperparameters used in the model may also change over time. This is a natural consequence of real-world changes, for example evolving responses from energy users as incentivisation of domestic flexibility becomes more common. We also expect that data collection improvements may unlock new features for use in modelling – such as if regularly updated participant counts become available.

3.4 Model performance

We give detailed numerical scores from evaluating the AFM and EDM below, along with the insights our modelling techniques give about the relative importance of model features. The key points from this analysis are:

- The AFM and EDM both perform better than the naïve forecast for almost all quantiles and flex directions (this is measured by the scaled quantile loss). They also generally have good coverage rates, especially for the GSP-level models.
- The AFM performance is stronger than that reported at a previous internal milestone (referred to as 'M5-3'). This is due to the increased data available and the use of a new model architecture, although the way the method for splitting training and test data has also been updated since.

- Forecast demand, GSP ID, temperature, and temporal variables (hour of day, day of week etc.) consistently have high feature importance for all models, with target flex also appearing for the EDM.
- Target flex is a proxy variable that cannot be fully simulated in these trial events. Its importance means BAU data (where this has a realistic definition) will be very valuable in future model development. Some retuning will likely be necessary once there is a significant quantity of BAU data.
- The aggregated regional models ('DFS' and 'LCM') perform worse than the GSP-level models. This is likely because they are trained on fewer data points, with just one data point per settlement period and DSRSP rather than also per GSP (~300 times fewer data points), which outweighs any improved predictability one might expect from averaging over a wider area. Aggregating data is also a complex process and will need to be handled with care as more DSRSPs participate in domestic flexibility.
- The EDM performs better at making in-event predictions than forecasting demand shift into event shoulders. The EDM has less training data available than the AFM, which will be a contributing factor, but we recommend alternative modelling approaches are considered once BAU data starts to become available.

AFM

Feature importance

Table 2 ranks the top six features for each model and flex direction, combined over quantile. The LightGBM model does not output coefficients in the same way as the linear model used for the M5-3 milestone. Instead, it uses the fitting algorithm to assess the importance of each input in contributing to the prediction (although not whether it was linked to higher or lower predicted flexibility). We do not give numerical values, as they are not directly comparable between models; instead, we look at the most important features for each model type and how often they appeared for each quantile, noting that the top feature sets may vary between quantiles.

The results are similar to those observed for the linear model, with GSP ID, demand, and temporal features consistently scoring highly. Unlike the previous evaluation, the dominant weather feature is temperature – this is what was expected when we originally selected weather features. The only unusual aspect is the appearance of forecast lead time in the OVO availability models, albeit with a far lower importance than that seen of the top three inputs. This is something that should be investigated in more detail. It is expected that lead time can help characterise the uncertainly in the demand forecasts to inform the upper and lower quantiles. In these models, it is also contributing towards the median prediction. Our suspicion is that it may be correlated with temporal variables due to the frequency with which demand forecasts were updated. There were also forecasts from the winter 2024 trials with longer than expected lead times. The model may be using this as a label to identify some common trends amongst events where this occurred, rather than something generalisable to future flex events. We recommend experimenting with

feature selection on a per-quantile basis to potentially omit lead times from the features used by the median models.

	GSP ID	Forecast demand	Hour of day	Day of week	Month	Temperature	Forecast lead time
Ohme Availability Turn-up	1	3	2	5	6	4	
Ohme Availability Turn-down	1	3	2	4	5	6	
OVO Availability Turn-up	1	3	2		4	5	6
OVO Availability Turn-down	1	2	3		6	5	4
OVO Utilisation Turn-up	1	2	5	4	6	3	
OVO Utilisation Turn-down	2	1	6	4	5	3	

Table 2: Top features ranked by overall importance for the three trained AFMs (Ohme-availability, OVO-availability, and OVO-utilisation), where 1 indicates the most important feature.

Performance results

When training the final models, we experimented with subsets of data (just as for the M5-3 milestone). Here, we found that the best performance for the utilisation model came from excluding the eight 'critical down' events from the training and test sets, giving better coverage rates with only a small negative impact on scaled quantile loss (SQL)[20]. This is not unexpected, as the presence of those events changes the shape of the flex distributions significantly. The trained availability models use all available data.

Table 3 shows the results for evaluating the Ohme availability 'GSP' model ('GSP' refers here to the model region as discussed in the Model regions section). Here, we calculate metrics for predictions on individual GSPs – this is in contrast to the UI functionality, where only the national total is displayed. This model is performing well, with the metrics calculated from evaluation on a broad set of data covering many months of behavioural variation. The scaled quantile losses are all less than (or equal to) one, indicating that the AFM is outperforming the naïve forecast. The turn-up models have better scores than those seen in the M5-3 milestone, this is likely due to the change in model type, as well as the larger amount of training data. The turn-down SQL figures

[20] This differs from the M5-3 milestone in which we got the best results by excluding the 'wrong way' events from the utilisation training and test sets. In both instances, we also compared models trained with no excluding of data.

are similar[21]. The model has good coverage rates (agreement between predicted quantiles and quantiles in evaluation data – see 3.3 Training and validation strategy), although some are slightly further from ideal values than those seen at the M5-3 evaluation. This could be because differences in participant behaviour between the winter 2024 and summer 2025 trials mean that the distributions for each trial are not quite the same making modelling both simultaneously more difficult. The M5-3 evaluation also used a different method to select the test set, so the scores are not directly comparable.

Metric	Turn-down					Turn-up				
	0.95	**0.75**	**0.5**	**0.25**	**0.05**	**0.05**	**0.25**	**0.5**	**0.75**	**0.95**
Coverage Rate	0.97	0.78	0.51	0.28	0.06	0.05	0.25	0.54	0.76	0.96
Quantile Loss	0.27	0.77	0.99	0.85	0.32	0.46	1.45	1.88	1.59	0.59
Scaled Quantile Loss	0.36	0.55	0.74	0.93	1.00	0.88	0.98	0.86	0.72	0.57

Table 3: Performance metrics for Ohme-availability 'GSP' AFM.

Table 4 shows the scores for the AFM trained on OVO availability data. These also have scaled quantile losses consistently less than one. As well as all performing more strongly than the naïve forecast, they also have better turn-up performance than the M5-3 model. As with the model trained on Ohme data, the turn-down results are comparable to those seen previously. We see that the coverage rates are not as close to the ideal values as those in Table 3, although they are nevertheless better than what was seen in previous models. The deviations are mostly in the turn-down models. We know from Figure 11 that parts of these distributions are complex, and from Figure 13 that the delivered flexibility values at the GSP-level have a lot of variation.

Metric	Turn-down					Turn-up				
	0.95	**0.75**	**0.5**	**0.25**	**0.05**	**0.05**	**0.25**	**0.5**	**0.75**	**0.95**
Coverage Rate	0.98	0.84	0.55	0.24	0.04	0.04	0.23	0.57	0.80	0.96
Quantile Loss	0.20	0.53	0.69	0.61	0.24	0.30	0.94	1.28	1.15	0.45
Scaled Quantile Loss	0.37	0.59	0.83	0.93	0.78	0.82	0.93	0.78	0.57	0.39

Table 4: Performance metrics for OVO-availability 'GSP' AFM.

[21] Note that the labelling of the turn-down quantiles is inverted here as compared to what was presented in the M5-3 milestone. Throughout this report, larger quantiles indicate more flex delivered *in the requested direction*. The details around this change are explained in Appendix C: Model implementation details.

Table 5 shows the performance metrics for the 'GSP' AFM trained on utilisation data. This model has far better coverage rates than those seen in the M5-3 milestone, where we observed consistent distortion in the shapes of the quantiles predicted by the model as compared to the data. As well as improvements from the model type, this is like affected by the decision to change how the evaluation holdout set was calculated. The set used for evaluation here is less affected by any seasonal variation (which may have affected the accuracy of the P376 baseline). The scaled quantile losses are again all less than one. The turn-up scores in particular are stronger than those seen before. The turn-down scores are slightly weaker, especially for the 0.5-0.95 quantiles. This could be down to the low amounts of delivered flexibility seen in the summer 2025 trial (as shown in Figure 10).

Metric	**Turn-down**					**Turn-up**				
	0.95	**0.75**	**0.5**	**0.25**	**0.05**	**0.05**	**0.25**	**0.5**	**0.75**	**0.95**
Coverage rate	0.94	0.74	0.48	0.24	0.05	0.04	0.21	0.45	0.72	0.93
Quantile loss	0.44	1.20	1.45	1.14	0.41	0.42	1.34	1.90	1.71	0.72
Scaled quantile loss	0.60	0.82	0.90	0.80	0.57	0.83	0.85	0.73	0.57	0.41

Table 5: Performance metrics for OVO-utilisation 'GSP' AFM.

We also evaluated the performance of the aggregated regional models. Here, we give average scores across all three AFM types (taking the mean of the scores seen for Ohme, OVO, availability, and utilisation). We do not give quantile loss scores here as they cannot be compared between the regional models. The results are shown in Table 6. We see that the 'DFS' and 'LCM' models both have worse coverage rates than the 'GSP' models. This is potentially due to the comparatively small amount of data used in training – aggregating over GSP means that some of the models had only around 1000 data points used to train, as compared to over 300,000 for the 'GSP' model[22]. The scaled quantile loss scores are more mixed. The aggregated models ('DFS' and 'LCM') are typically better at predicting the values of the extreme quantiles (0.95 up/down) and median but worse at the near quantiles (0.05 up/down), with some of the SQL values indicating that the aggregated models are performing worse than the naïve forecast used for comparison. As more data is collected during BAU, the aggregated models will want to be analysed carefully and likely rebuilt. The poorly performing quantiles are those which are of most interest when making risk-aware decisions. They will also benefit greatly from increased training data and a more careful treatment of participant number variation in each GSP.

We saw the same trend throughout in that performance (as measured by SQL) is generally best for the 0.95 quantiles and poorer for quantiles in the 0.05-0.5 range. This potentially makes applying these predictions for planning in worst-case scenarios (where flexibility is requested but

[22] The rows for each of around 300 GSPs are summed to give a single row describing nationwide data.

does not materialise significantly) tricky. This is likely down to the shapes of the distributions as seen in Figure 11 – there is more noise associated with small values of delivered flexibility.

Metric	Region	Turn-up					Turn-down				
		0.95	0.75	0.5	0.25	0.05	0.05	0.25	0.5	0.75	0.95
Coverage rate	GSP	0.97	0.79	0.51	0.25	0.05	0.04	0.23	0.52	0.76	0.95
	LCM	0.94	0.74	0.63	0.43	0.22	0.06	0.27	0.47	0.72	0.86
	DFS	0.96	0.74	0.67	0.38	0.16	0.15	0.28	0.53	0.70	0.86
Scaled quantile loss	GSP	0.44	0.65	0.82	0.89	0.78	0.84	0.92	0.79	0.62	0.46
	LCM	0.38	0.55	0.63	0.83	1.25	0.86	0.64	0.53	0.47	0.39
	DFS	0.30	0.49	0.52	0.72	1.01	1.03	0.61	0.52	0.48	0.53

Table 6: Performance metrics for regional AFMs, averaged over individual DSRSP and trial models.

EDM

Feature importance

In order to evaluate the performance of the EDM, models were trained on the data from the summer 2025 trial, as this data includes the six-hour shoulders required to estimate demand shift. For the availability trials, this data included 1045 settlement periods for turn-up events, and 523 for turn-down events, whereas for utilisation there were 1216 turn-up event settlement periods and 308 settlement periods for turn-down events.

Table 7 ranks the top six features for each model and flex direction, combined over quantile. As with the AFM, the top six features are given for each. The EDM treats DSRSPs as equivalent, and thus the availability model includes data from both OVO and Ohme in a single model, whereas the utilisation model only includes OVO data.

Similarly to the AFM, GSP ID, forecast demand, and hour of day are important features in both the availability and utilisation models, and temperature is the dominant weather feature. Target flex appears in the top six features by importance for all models except utilisation turn-up, where it was the eighth most important feature. Target flex appears as an important feature in all models, despite the fact that the target flex used here was a proxy provided by the 50% quantile from the AFM. We expect that a true target flex would provide a stronger signal and therefore be a more important feature. Thus, we strongly recommend that accurate target flex data should be collected to be used for training the models. For availability turn-up and turn-down, and utilisation turn-up, target flex scored moderately in comparison to the top features for each model, whereas for utilisation turn-down, target flex appeared almost as often as GSP ID which was the top feature.

	GSP ID	Forecast demand	Target flex	Hour of day	Day of week	Temperature	Time through shoulder
Availability Turn-up	1	2	3	4	5	6	
Availability Turn-down	1	2	5	3	4	6	
Utilisation Turn-up	2	1		5	3	6	4
Utilisation Turn-down	1	3	2	4		5	6

Table 7: Top features ranked by overall importance for the EDMs, where 1 indicates the most important feature.

Performance results

The summer 2025 data used to train the EDM did not include any 'critical down' events, and therefore these events did not impact the EDM evaluation. We experimented with excluding wrong way events from training the utilisation model, however this resulted in too little data remaining to evaluate the turn-down model and had little impact on the evaluation scores for the turn-up model.

Table 8 shows the results for evaluating the 'GSP' availability model on Ohme data. The scaled quantile losses for turn-up are less than one, which indicates that the model performs better than the naïve forecast. However, the scaled quantile losses for the central quantiles of the turn-down model are greater than one. This is in contrast to the AFM, where the scaled quantile loss for the Ohme availability 'GSP' model was lower for the median than the 0.05 quantile. More data, and accurate target flex data may improve this. Similarly, the coverage rates are close to the quantile values for the turn-up model and are far from the quantile values for the 50% and 75% quantiles for the turn-down model. Similarly, Table 9 shows such results for OVO data, and the scaled quantile loss and coverage values are typically worse for the turn-down model than for turn-up. Finally, the same pattern is shown in Table 10 for the combined DSRSP data.

Metric	**Turn-down**					**Turn-up**				
	0.95	**0.75**	**0.5**	**0.25**	**0.05**	**0.05**	**0.25**	**0.5**	**0.75**	**0.95**
Coverage rate	0.93	0.57	0.39	0.24	0.06	0.06	0.24	0.49	0.77	0.96
Quantile loss	0.49	2.35	3.16	1.96	0.64	0.53	1.59	2.02	1.70	0.61
Scaled quantile loss	0.79	1.56	1.60	1.01	0.71	0.65	0.86	0.93	0.87	0.72

Table 8 : Performance metrics for the 'GSP' availability EDM on Ohme data.

Metric	Turn-down					Turn-up				
	0.95	**0.75**	**0.5**	**0.25**	**0.05**	**0.05**	**0.25**	**0.5**	**0.75**	**0.95**
Coverage rate	0.90	0.65	0.42	0.26	0.06	0.05	0.25	0.48	0.72	0.94
Quantile loss	0.34	1.02	1.38	1.16	0.43	0.34	1.04	1.42	1.24	0.46
Scaled quantile loss	0.72	0.93	1.04	0.91	0.71	0.73	0.90	0.96	0.84	0.66

Table 9: Performance metrics for the 'GSP' availability EDM on OVO data.

Metric	Turn-up					Turn-down				
	0.95	**0.75**	**0.5**	**0.25**	**0.05**	**0.05**	**0.25**	**0.5**	**0.75**	**0.95**
Coverage rate	0.93	0.63	0.42	0.25	0.07	0.05	0.24	0.49	0.76	0.96
Quantile loss	0.65	2.53	3.25	2.26	0.74	0.65	1.99	2.55	2.14	0.77
Scaled quantile loss	0.81	1.29	1.28	0.94	0.67	0.70	0.88	0.92	0.83	0.70

Table 10: Performance metrics for the 'GSP' availability EDM on 'All' DSRSPs.

Exploratory analysis suggests that models trained on one DSRSP may yield better results. However, combining separate predictions from different DSRSPs has the same challenges as combining predictions from different GSPs (see Model regions in Section 3.2 for a technical description). Therefore, to produce valid predictions for combinations of DSRSPs, we train a single model on the combined data. Training a model in this way requires assuming the DSRSPs are equivalent, this is not a realistic assumption, as different DSRSPs deliver different distributions of flex (Figure 11) due to differing demographics or dispatch techniques. To improve this, we recommend that DSRSPs should be implemented as a feature to the models, each GSP ID and DSRSP ID combination is target encoded, or that alternative model architectures are considered (see Section 4.4 Model improvements).

In contrast to the availability model, the utilisation model, which was only trained on OVO data, performs better than the naïve forecast for all directions and quantiles, as shown in Table 11. Additionally, the coverage rates are consistent for both turn-up and turn-down, despite the small sample size. This may be because seasonal variability across the summer 25 trial was less pronounced, which would affect the levels of flexibility delivered and the P376 baseline calculation.

Metric	Turn-down					Turn-up				
	0.95	**0.75**	**0.5**	**0.25**	**0.05**	**0.05**	**0.25**	**0.5**	**0.75**	**0.95**
Coverage rate	0.94	0.74	0.52	0.29	0.08	0.07	0.28	0.53	0.76	0.95
Quantile loss	0.14	0.44	0.58	0.5	0.2	0.32	0.95	1.2	0.98	0.34
Scaled quantile loss	0.71	0.88	0.93	0.85	0.72	0.51	0.75	0.81	0.7	0.48

Table 11: Performance metrics for the 'GSP' utilisation EDM on OVO data.

Table 12 shows the performance of the aggregated regional models. We give the average coverage rates and scaled quantile loss scores across both the utilisation and availability EDMs. As with the AFM, the 'DFS' and 'LCM' models have worse coverage rates than the 'GSP' model. The scaled quantile loss scores are more varied. The 'GSP' model performs well on turn-up events, and poorly for the centrals on turn-down events. However, 'DFS' turn-up shows improved scaled quantile loss compared to the 'GSP' models, for all quantiles except 0.05, whereas the values are worse for turn-down, except for the 0.25 quantile where the difference is negligible. This is in opposition to the AFM, which performs well on extreme quantiles and worse on conservative ones in general (Table 6). The 'LCM' models consistently have higher scaled quantile loss values for both models except for turn-up 0.75 where the scaled quantile loss is slightly smaller for 'LCM' than 'GSP'. As with the AFM, this is likely to be due to small amounts of data available for the aggregated models.

Metric	**Region**	**Turn-down**					**Turn-up**				
		0.95	**0.75**	**0.5**	**0.25**	**0.05**	**0.05**	**0.25**	**0.5**	**0.75**	**0.95**
Coverage rate	**GSP**	0.92	0.65	0.44	0.26	0.07	0.06	0.25	0.50	0.75	0.95
	LCM	0.87	0.50	0.45	0.31	0.17	0.20	0.48	0.64	0.77	0.91
	DFS	0.77	0.53	0.40	0.41	0.23	0.16	0.38	0.45	0.64	0.86
Scaled quantile loss	**GSP**	0.76	1.16	1.21	0.93	0.70	0.65	0.85	0.9	0.81	0.64
	LCM	1.11	1.47	1.97	1.05	1.13	0.94	0.96	0.99	0.76	0.93
	DFS	1.06	1.31	1.55	0.92	0.91	0.87	0.83	0.74	0.66	0.64

Table 12: Performance metrics for each EDM region, averaged over the trial models.

Finally, Table 13 shows the performance metrics for the 'GSP' availability EDM on Ohme data, for within event and within shoulder settlement periods. Overall, the model performs better within the event than in the shoulder, which is likely due to a weaker signal to noise ratio in the shoulders. More data may improve this. For each event there are 12 hours of shoulder (for example, for availability turn-up, there are 88 settlement periods within event, and 1045 settlement periods when including the shoulder), and therefore the shoulders contribute more to the model than the events do. Therefore, we recommend weighting the event and shoulder data to mitigate this. Additionally, we recommend exploring the impacts of implementing separate models for event and shoulder periods.

Metric	Event or Shoulder	Turn-down					Turn-up				
		0.95	**0.75**	**0.5**	**0.25**	**0.05**	**0.05**	**0.25**	**0.5**	**0.75**	**0.95**
Coverage rate	**Event**	0.91	0.65	0.47	0.24	0.05	0.04	0.25	0.52	0.76	0.95
	Shoulder	0.93	0.57	0.39	0.24	0.06	0.06	0.24	0.49	0.77	0.96
Scaled quantile loss	**Event**	0.54	0.69	0.80	0.99	1.03	0.88	0.92	0.75	0.59	0.44
	Shoulder	0.81	1.61	1.63	1.01	0.70	0.61	0.85	0.98	0.97	0.84

Table 13: Performance metrics for the 'GSP' availability EDM on Ohme data for settlement periods in events and in shoulders.

These findings should be interpreted with caution as the target flex these models were trained and evaluated with is provided by the 50% quantile of the AFM as a proxy. We do not expect this proxy to accurately represent the values of target flex used in BAU, and thus these results may not reflect the performance expected in BAU.

3.5 User interface

A graphical user interface (UI) helps users make intelligent decisions based on the output of the models. The use cases and requirements for the UI, were identified and gathered through a number of workshops with future NESO users. These outputs, combined with ongoing design feedback from said future NESO users, have been incorporated into the UI.

The UI allows the user to construct a request for either the AFM or EDM and display the resulting predictions on a graph. Figure 22 shows a sample output for the AFM. We wrote the UI application in TypeScript using the Remix framework. It is hosted in the Azure Web App service, which allows close integration with the remainder of our Azure infrastructure and provides security through easy access to Microsoft's Authentication Library.

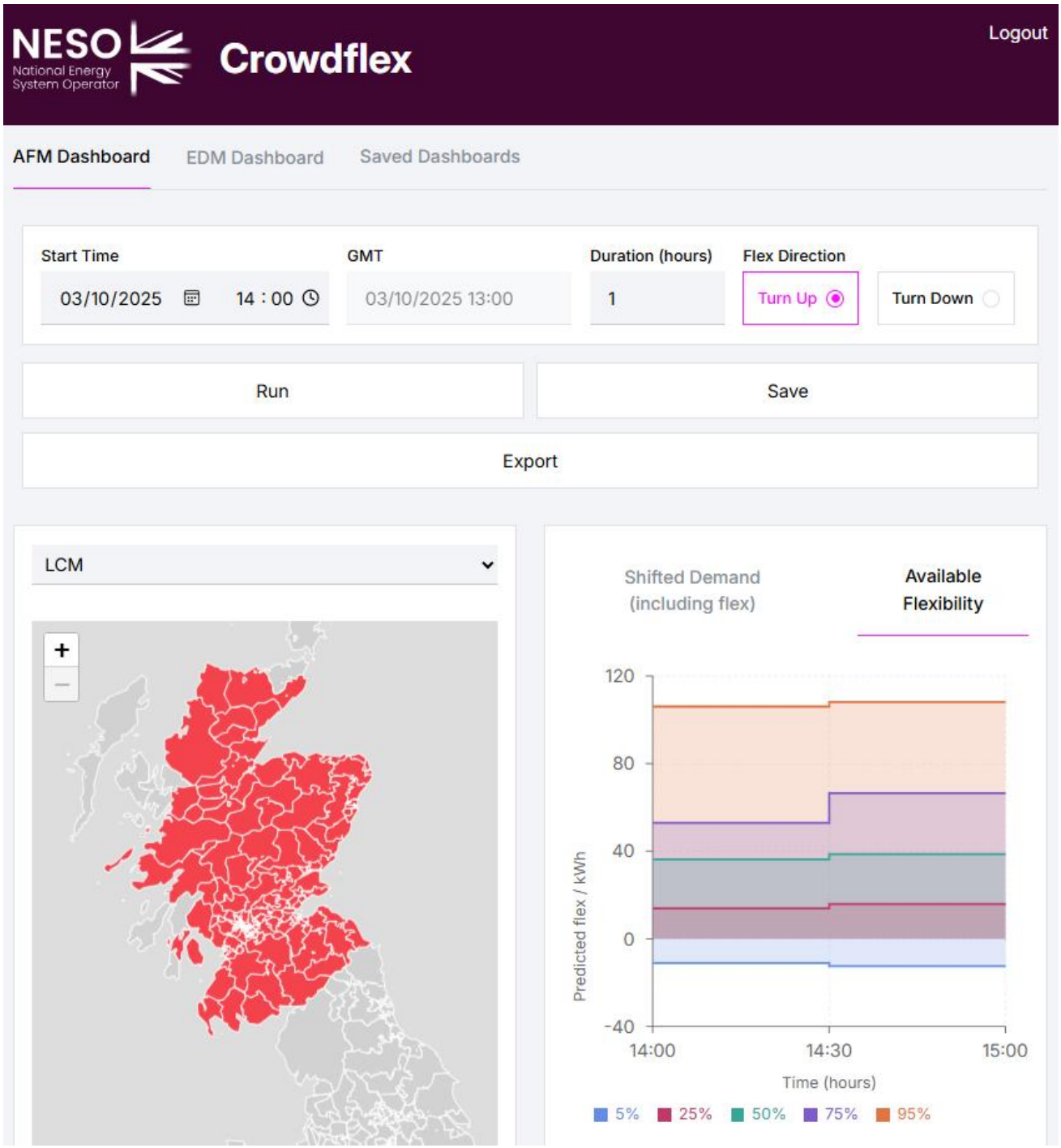

Figure 22: AFM user interface.

At the top of Figure 22, the user chooses whether they wish to query the AFM or EDM. Then, they input the start time (in local time with GMT displayed for clarity), the duration and flex direction of

the proposed flex event, select the region in the bottom-left with the aid of the map[23] and click 'Run'. On return, the graph in the bottom-right displays each quantile of the predicted available flexibility for each settlement period within the requested timeframe. These values are the total over the whole region (or sum of GSP predictions for the 'GSP' region). The alternative 'shifted demand' graph shows the quantiles of total predicted demand, consisting of the forecast demand (in black, labelled 'baseline' but not to be confused with actuals baseline used elsewhere in this report) plus the predicted flexibility, as shown in Figure 23. The user can hover over each data point to view a table of values, as demonstrated in the figure. They can also hide selected quantiles by clicking on them in the legend.

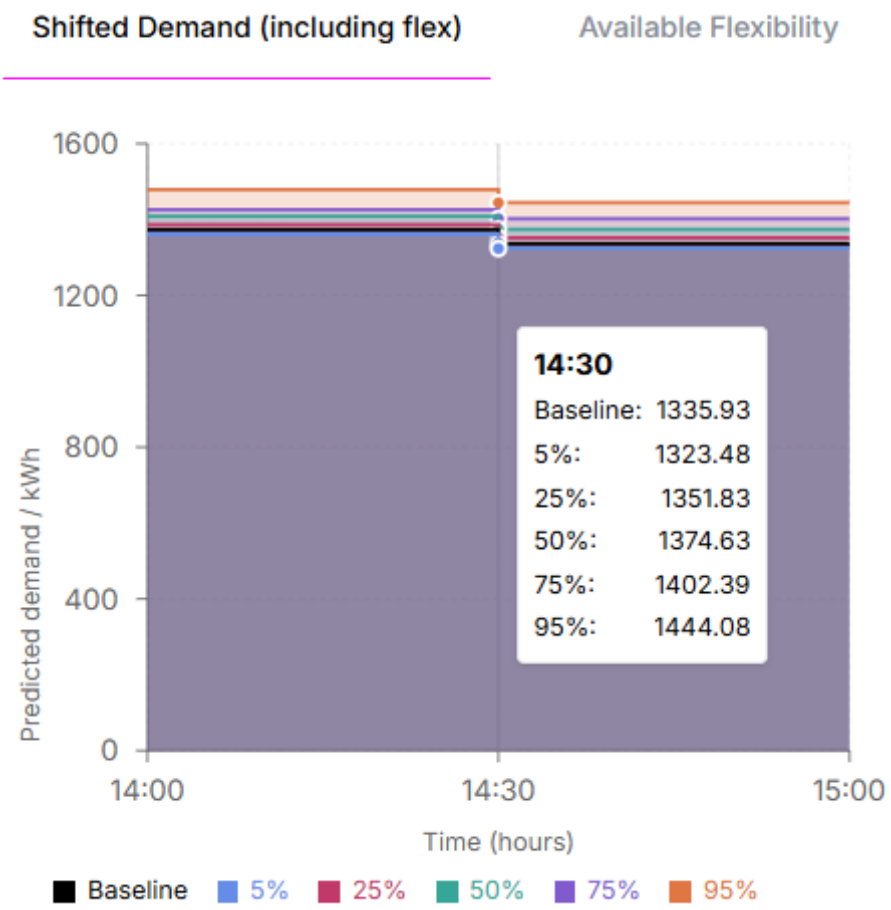

Figure 23: Shifted demand graph corresponding to the available flexibility graph shown in the AFM UI. Also showing table of values obtained by hovering over the data points on the graph.

[23] The user is currently unable to select individual GSPs to query using the UI. Instead, they will need to query the model API directly using the 'gsp_list' parameter and process the JSON response themselves.

To save the data for later or for ingestion into another tool, clicking the 'Export' button downloads a JSON file containing all the predictions. For the 'GSP' region, this also includes predictions and forecast demands for each GSP individually. Alternatively, the user can save this UI configuration by clicking the 'Save' button, recovering it later from the 'Saved Dashboards' tab.

The EDM UI is very similar, as shown in Figure 24. The only differences are an input box for target flexibility (which may be left empty during trials to query the AFM as a proxy) and a dropdown to select which DSRSPs contribute towards that target flex. The output graphs include the six-hour shoulders either side of the event. The chosen start time refers to the start of the event, not the shoulder.

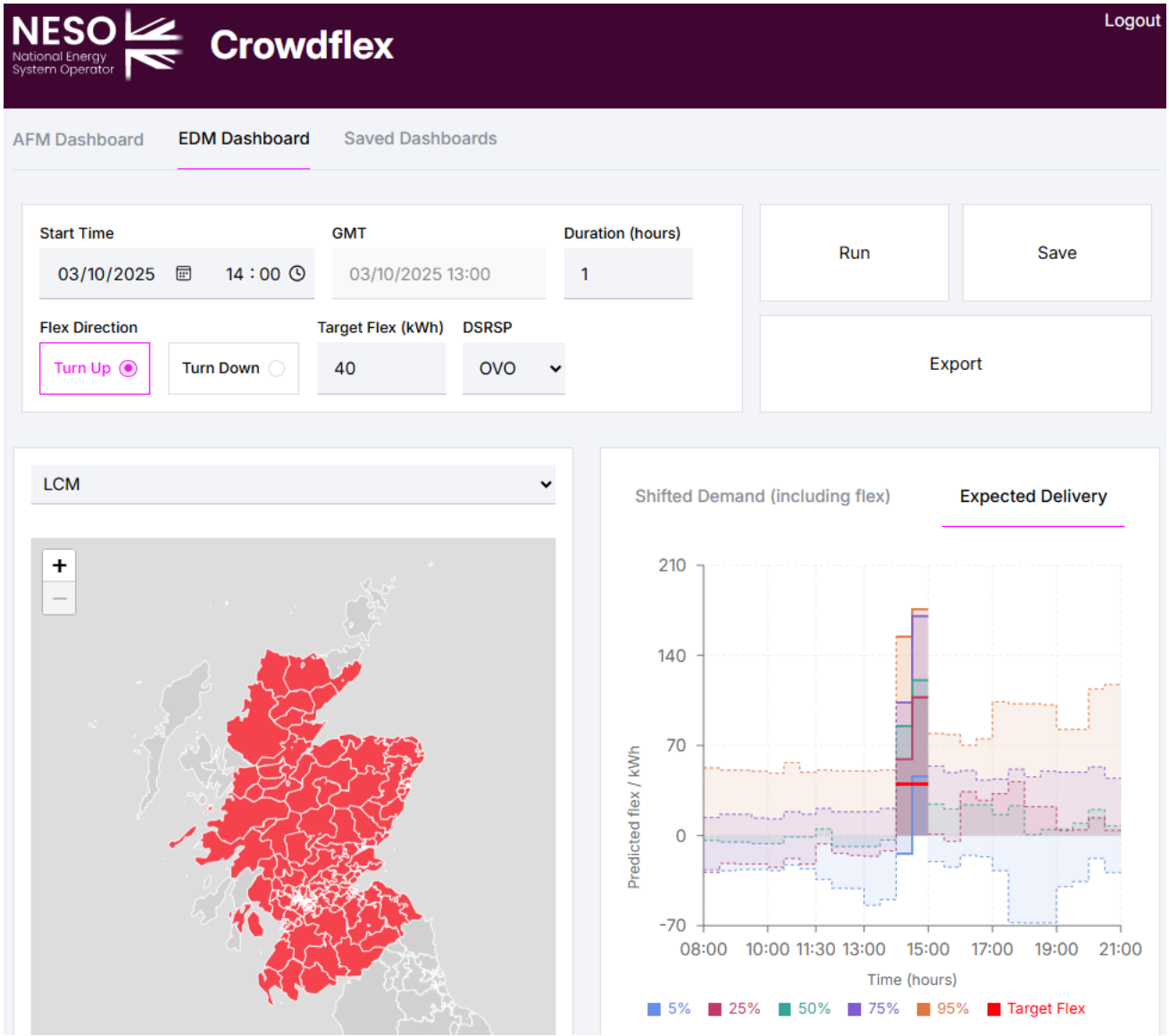

Figure 24: EDM user interface.

4. Challenges and learnings

4.1 Data-related challenges

Over the course of the project, challenges arose due to the high volumes of data required, the evolving nature of the trials, and the need for continuous, reliable delivery.

We originally expected that DSRSPs would update their forecasts throughout the day, so we queried hourly to ensure we had the most up-to-date forecasts. In practice, the DSRSPs were only able to update their demand forecasts daily (primarily limited by delays in which they received data from meters). This meant that our earlier collection had a lot of duplicated data – 24 hourly copies of each daily forecast. This duplication was handled automatically by our pipelines, but we later decided to only collect demand forecasts daily to reduce the volumes of data we needed to handle (speeding up processing times and reducing compute costs). This is an aspect that will need to be reconsidered going into BAU. If data is only collected daily (and the sources it pulls from also updated daily) then in a worst-case scenario it may have a 48-hour latency, even before processing times are factored in. NESO will need to agree with DSRSPs the frequency of their demand forecast.

Over the course of the trials, there were many difficulties due to changes in API behaviour and data formats which interfered with data collection. New issues were discovered with the APIs in each new trial: regression of previous bug fixes, insufficient schema validation, and differing interpretations of the evolving API specification. Rather than acquiring the data regularly in small numbers of queries as intended, much of the data was collected in large 'backfill' requests, which introduced significant delays to data collection. This highlights the need for ensuring that APIs and data formats are as stable as possible during BAU data delivery. Sufficient time and care should be allotted to align requirements between DSRSPs and NESO and to implement comprehensive testing.

Due to this project involving multiple distinct trials, many APIs were created, generating complexity when querying for data. We recommend that a more streamlined process is put in place for BAU data delivery, preferably with a single API endpoint per DSRSP.

Confusion over the API specification is something that was gradually resolved over time. The DSRSPs and Smith Institute had different requirements and uses for the data that was being handled – most notably around the 'generated time' attached to each forecast. Adding clearer explanations of our intended modelling process and some worked examples of sample API responses would likely have streamlined this process – we recommend adding explicit examples to future versions. Additionally, the risk of miscommunication could be mitigated through a dedicated Q&A session between the partners to ensure each specification has a common understanding before work commences.

Weather forecasts were not collected live for several months in parts of the summer 2024 and winter 2024 trials due to the infrastructure problems described in the technical challenges section below. Unfortunately, the Met Office spot API only provides the latest forecast. Instead, we

backfilled these from an archive of their UK atmospheric forecast[24] hosted on AWS. Since this is a different forecast product, variables, definitions, forecast locations and forecast horizons are all slightly different, which degraded data consistency. Ensuring the reliability of continuous data delivery and storage will be critical during BAU to avoid unrecoverable data loss.

4.2 Technical challenges

Development environment uncertainties

The original plan had been to develop and train the CrowdFlex models within the newly introduced Advanced Analytics Environment (AAE) system hosted by NESO. The AAE is a pre-configured Azure Machine Learning workspace, which is intended to be swiftly set up and deployed. Although NESO set up the AAE quickly, later efforts revealed several inconsistent security policies. These policies, inherited from the higher-tier National Grid Azure subscription, proved both complex and time-consuming to resolve between Smith Institute and NESO.

Since CrowdFlex was the first project to make full use of the new platform, there were no proven solutions to guide the team through the challenges being encountered. This led to a long period of uncertainty as to which platform the models would be developed on and as a result work on the AFM model stopped periodically. To minimise the delay in model development, a decision was taken to transition to Smith Institute's Azure environment for the remainder of the project. This decision was only reached once flexibility trials had already been underway for over two months. In the interim period, we had to develop temporary data ingestion pipelines to ensure that we did not miss critical data.

Smith Institute welcomes NESO efforts to make the delivery of software by external suppliers more streamlined, whether through the AAE or otherwise. However, such tools and processes should be thoroughly tested and trialled to ensure they are fit for purpose before they are used in practice.

Transfer back to NESO estate

Developing the models on the Smith Institute's Azure environment has resulted in a requirement to transfer the model code and data back to NESO. After evaluating the available options, it was decided to deploy the models on a dedicated Azure Landing Zone (ALZ) rather than the AAE. The ALZ offers greater flexibility, enabling the optimization of Azure resources and the customization of the environment to suit specific NESO needs, in addition to allowing NESO to explore other data sources and prepare potential changes to the model.

Using ALZ reduces susceptibility to disruptions arising from the ongoing IT separation between NESO and National Grid. This separation had previously caused permission issues on the AAE,

[24] https://aws.amazon.com/marketplace/pp/prodview-oiodcatwsyjwm?sr=0-1&ref_=beagle&applicationId=AWSMPContessa

which impacted early project development. By choosing the ALZ, the risk of encountering similar permission-related challenges is significantly minimized.

NESO IT Architects, Business Partners, and the designated model owner are actively parallel planning the transition to BAU. A critical component of this planning process is identifying the existing NESO systems—such as PEF, OBP, and demand forecasting—with which the models will need to integrate.

LQR model training times

Training the machine learning models in this project requires fitting the models to the training data. As the quantity of data increased through the winter 2024 trial, the fitting time for the initially implemented linear quantile regressor (LQR) models became large. To reduce this, we integrated the Gurobi solver in place of scikit-learn's inbuilt use of the HiGHs open-source solver. This reduced fitting times considerably but fitting times nevertheless became prohibitively long as the volume of data continued to increase. Further, use of Gurobi requires a license subscription which added an additional cost and deployment requirement for NESO. Ultimately, we found that switching to gradient-boosted tree (GBT) models not only produced more reliable and accurate forecasts, but reduced training time to more manageable levels. While the implementation of the LQR models remains in the codebase, we recommend the GBT models are used for BAU.

Azure Data Factory

We chose Azure Data Factory (ADF) a reliable, scalable data ingestion and transformation platform, which we could deploy easily and integrate well with other Azure components (chiefly Azure Machine Learning). ADF provides a reliable and effective data ingestion and transformation system. Once the data ingestion pipelines were set up, the vast majority of pipeline failures were due to data failing validation (see 4.1 Data-related challenges) rather than any kind of infrastructural failures. In addition, scaling up compute for our data transformation pipelines, which will likely be necessary in BAU due to increased data volumes, is simple. We also made use of CI/CD practices and deployed two near-identical copies of our data factory, with one copy used for development and one for live data processing; while ADF being based on static configuration files causes some friction in development, it does make this kind of development-production separation relatively simple.

However, use of ADF does have some drawbacks. The user documentation is generally of low quality – many features of ADF as well as the underlying domain-specific languages were poorly documented, which makes development and troubleshooting harder.

ADF's graphical user interface (GUI)-based development interface (ADF Studio) causes a lot of friction for collaborative development. Under the hood, ADF uses automatically-generated code files, which are usually hidden from developers by the GUI. However, because the Git integration built into ADF Studio is extremely basic, these automatically-generated files were the only way to review code changes. This disconnect between development and review also caused headaches

when multiple developers made changes simultaneously, as the resulting conflicts were difficult and time-consuming to resolve, requiring manual intervention with external editors.

Testing and debugging were also a challenge. ADF Studio has a limited ability to pass sample data through data pipelines for testing and debugging purposes, but it was often difficult to isolate specific parts of the data that were causing problems, and in some cases the interface would crash due to our input datasets being too large. In the same vein, while ADF provides the ability to create reusable components (called 'flowlets') that can be used in multiple data flows, testing those components in detail is nearly impossible due to ADF Studio's limited debugging capabilities. With all of these issues, there was often no practical way to test changes except by deploying them to our live data pipelines, which risked disrupting our data collection.

4.3 Collaboration and coordination challenges

There have been a number of learnings from working a large scale, fast paced multi-year and multi-party project that can be utilised for future projects. The consortium is now a strong and effective working group having learnt and adapted to these challenges over the lifecycle of the project.

At the start of the project, it was identified that, to utilise domestic flexibility, both the AFM and EDM were required. The AFM was initially designed and developed and the EDM remained a future opportunity. While this meant that when the EDM, that was commissioned in March 2025, could be developed faster with the application of the lessons from the AFM development, it did reduce the development cycle and limited the data set for training to only the summer 2025 trials.

Octopus Energy was initially part of the consortium as a key member. However, they withdrew from the project early in the first quarter of 2024. This withdrawal had a notable impact on the volume of trial data that could be used for model training and ultimately an impact on the forecasting performance of the models. The addition of more DSRSP data and other data sources in future developments will strengthen the model forecasts.

Inevitably on multi-year projects, staff turn-over is a feature which can lead to a reduction in pace while new staff learn the ropes. Both Smith Institute and NESO realised this and project teams implemented measures to keep turnover of project team members low, which helped to maintain a good collaborative working relationship on the project while maintaining progress, output and deliveries.

Similarly, where a project involves a high number of participant organisations, communication challenges can occur in both the technical and administrative domains. The fast-paced nature and scale of CrowdFlex made this especially challenging but critical. Some early differences in interpretation and expectations did lead to a degree of inefficiency on CrowdFlex but this was identified and improved. These issues highlighted the importance of alignment regarding expectations through agreed written channels and provision of adequate notice of desired outcomes.

Data sharing between parties required a dedicated data sharing agreement to be established to ensure appropriate use of the consumer data and compliance with relevant standards. While

ultimately necessary, this process was initiated late in the project and proved time-consuming. Earlier identification of this requirement would have allowed for smoother integration and reduced delays.

A recommendation we would make going forward is to identify and include input from a defined "model owner" within NESO. Our experience has shown that the effectiveness of mathematical modelling is improved by engaging with a BAU user who collaborates on design and build through immediate provision of use cases and user requirements. For the EDM in particular, this view would have been a useful voice to create further alignment, simplifying and speeding up the process through a clearer understanding of the final user interface requirements.

4.4 Model improvements

There are several feature ideas for the models that were considered for development but were not taken forward.

Target flexibility

During the trials, target flexibility was not recorded, as DSRSPs were attempting to deliver the maximum possible flexibility. In development, we used predictions from the AFM as a proxy for target flexibility input to the EDM, however we do not expect that this proxy should accurately reflect the target flexibility in BAU. Therefore, we recommend that accurate target flexibility data should be collected in BAU, and once enough data has been collected, this training data should be discarded.

Event interdependence

Both models consider dispatch events in isolation. They do not account for either consumer fatigue in the case of mainly manual flexibility, or the time taken for automated flexibility to become available again (e.g. because the EVs are fully charged). We expect this limitation to lead to over-forecasting, and recommend that it be addressed as part of ongoing model development.

Participant counts

Over time, if more individuals become participants of flex events, we would expect available flexibility to increase and speculate that expected delivery may increase in reliability. Therefore, including participant counts for each GSP as a feature in the model may mitigate model drift due to this.

Quantile crossing

With all of our quantile model types, each quantile is fitted independently, leading to the possibility of quantiles crossing. Although as yet unobserved, we may wish to implement a guard to keep quantiles monotonic.

Special events

Special events, such as bank holidays or significant sporting events, are likely to impact both available flex and expected delivery. Different types of events are likely to have different effects and may coincide, and therefore they may need to be included as features individually.

Model regions

NESO may wish to add or change model regions in future. We recommend in the first instance defining more regions. However, should arbitrary combinations of GSPs be required, we suggest investigating copulas, a statistical method for modelling the relationship between random variables. A second approach could be bootstrapping, but we found it brought extreme quantiles too close to the median, as it assumes each GSP delivers independently of all others; this is the opposite problem to summing quantiles (the approach used for the 'GSP' region – see Model regions in section 3.2). A third candidate approach is a machine learning model (such as gradient-boosted trees) that captures the relationships between GSPs; this would likely be a less-engineered version of a copula. Alternatively, such approaches could be used to perturb a regional model if a small number of GSPs is to be added or removed, which may yield better results than aggregating over the 'GSP' region.

DSRSPs in EDM

We recommend adding encoded DSRSP ID as a feature of the EDM. We expect that each DSRSP may have different demographics and their delivered flex may depend on this. The total flex would also depend on the proportionate target flex and forecast demand of each DSRSP, which we may think of as arbitrary bid/offer combinations. Aggregating DSRSPs would then pose similar challenges to aggregating over GSPs, so the same techniques could be explored (see Model regions).

Notice period

While notice period was initially considered to be a candidate for model input, it was ultimately excluded. In BAU, NESO does not have visibility as to what notice period the DSRSPs offer customers ahead of a flexibility event. However, notice period does have an impact on flex delivery[25].

For NESO, the key point of interest is the time interval between the announcement and approval of bids or offers, and the actual start of said event. This is particularly relevant when considering timescales associated with initiatives such as Demand Flexibility Service (DFS) and Local Constraint Market (LCM). However, as there was no direct equivalent within the CrowdFlex trials,

[25] CrowdFlex report: Utilisation trial Winter 2025 section 4.5.1.1 Effect of notification period

this aspect was not explored during the project. Nevertheless, the inclusion of notice period remains a relevant consideration for future model development.

Archetypes

Different customer archetypes may exhibit varying behaviours during flexibility events. These behavioural differences have the potential to enhance the accuracy of forecasting related to the reliability of flexibility delivery. The task of archetyping CrowdFlex participants was allocated to a separate work package and as a result, archetype analysis was not incorporated into the core modelling activities.

Long-range forecasting capabilities

During the course of the project, the potential application of the developed models for long-range forecasting - specifically for several months or even years ahead - was considered. This enquiry focused on whether such forecasts could support Clean Power 2030 Action Plan (CP30) estimations and facilitate long-term energy supply planning. However, this request arose after the model specification had already been finalised and formally approved. Consequently, it was determined to fall outside the agreed project scope.

It is important to note that, while the models are technically capable of generating long-range forecasts provided that suitable input data is available, the underlying forecasting algorithms were not designed or optimised for such extended timescales. New models can be developed, with the appropriate data inputs, that could provide long-range forecasting that would be better suited for this use case.

Multi-day flexibility events for constraint management

The original use case for developing the models was assumed to be NESO constraint management. In discussions with the NESO constraint management team, it became evident that effective application in this area would necessitate the ability to forecast a sequence of planned flexibility events spanning multiple days. However, integrating this multi-day forecasting requirement into the trial planning process was not considered feasible. As a result, this approach was not pursued further during the course of the project.

4.5 Key takeaways for future projects

Throughout the course of the project, several important lessons were identified that are essential for the effective management of complex, multi-party initiatives. These insights are crucial for future projects seeking to enhance efficiency, resilience, and collaboration.

Data management proved to be a significant challenge throughout the project. Ensuring the stability and validation of APIs and data formats prior to the commencement of trials is imperative to prevent delays and avoid costly reprocessing efforts. For future initiatives, streamlining data delivery is recommended—this can be achieved by adopting a single API

endpoint per DSRSP and establishing a mutually agreed update frequency. Such measures will help to eliminate duplication and reduce latency.

Readiness of the technical infrastructure emerged as another key consideration. It is essential to validate development platforms before use to mitigate uncertainty and avoid delays. Contingency planning for hosting and data ingestion should be prioritised to minimise disruption. For future projects, technical strategies should be developed early that consider the requirements of the technical development.

Effective collaboration and governance require early alignment among all partners. Establishing clear expectations and robust communication protocols at the outset of a project is fundamental. Clearly defining roles, including appointing a dedicated model owner, ensures stakeholder alignment and clarity regarding requirements. Formal agreements, such as those concerning data sharing and compliance processes, should be initiated at the beginning of the project to avoid delays in integration.

Focused attention on communication and coordination is essential. The provision of written specifications, worked examples, and Q&A sessions can substantially reduce the risk of misinterpretation. Even when timelines are tight, it is important to allocate sufficient time for alignment to ensure cohesive project delivery.

5. Conclusions

The CrowdFlex project has demonstrated the technical feasibility and strategic value of integrating domestic flexibility into grid management for Great Britain. Through the collaborative efforts of a diverse consortium, the project delivered robust models, extensive trial data, and actionable insights for future deployment.

Smith Institute's contribution has delivered models that are able to forecast flex in changing circumstances and supporting infrastructure to enable NESO to integrate domestic flexibility into business-as-usual grid operations. The development and validation of the Available Flexibility Model (AFM) and Expected Delivery Model (EDM) provide probabilistic forecasts for domestic flexibility. We have integrated large-scale consumer trial data and weather forecasts into scalable data pipelines and modelling infrastructure.

During this work we have navigated challenges including:

- Data quality and completeness issues, driven by technical limitations, evolving APIs, and GDPR requirements for participant privacy. Reliable modelling depends on high-quality, representative, and timely data. Early investment in robust data validation and integration processes is essential.
- Infrastructure and development environment constraints, which impacted project timelines and required adaptive solutions. Technical infrastructure must be chosen and configured to support both experimentation and scalable deployment, with flexibility to adapt as project needs evolve.
- Coordination across multiple organizations, highlighting the need for clear governance, communication, and early data sharing agreements. Effective collaboration requires clear roles, responsibilities, and shared understanding of objectives and requirements.

The models and processes developed in CrowdFlex provide a strong foundation for BAU integration, but differences between trial conditions and real-world operations must be addressed. Ongoing model refinement and data collection will be necessary to ensure reliable and actionable forecasts.

CrowdFlex has advanced the understanding of domestic flexibility's role in grid management and laid the groundwork for future innovation. The project's insights and recommendations will inform ongoing efforts to decarbonize the energy system and empower consumers as active participants in the transition to a more flexible, resilient grid and aid progress towards the Government's Clean Power 2030 Action Plan (CP30).

Appendices

Appendix A: Data pipelines implementation details

Bronze

The pipelines save the collected data in JSON files in an Azure blob storage account. The DSRSP data directory structure is partitioned using event ID, DSRSP ID, meter-type, and the 'at time' (UTC timestamp) that the file was collected at. The weather data is partitioned using GSP ID and 'at time'. When planning for BAU, we recommend restructuring this data to rework the keys:

- Year/month: Adding extra temporal keys will make the data easier to manage as its volume grows, especially if a human needs to browse it manually. They can also be used to feed directly into the partitioning of data at silver and gold, instead of needing to derive new keys within those pipelines. These should be top-level keys.
- Event ID: After the transition to BAU, this will no longer be a meaningful label to apply to the demand forecast data – since we anticipate collecting it continuously. Instead, if collecting the data hourly, a 'day' key (representing the day of the month) is probably the best way to subdivide it.

The data is additionally written to a 'Delta Lake', storing the contents of each file as a single entry in the table. A Delta Lake is tabular in format with built-in compression and scanning capabilities (and a more efficient storage structure than a human-readable format like CSV). Delta Lakes also include look-back functionality, although we restricted some of this due to the frequency of read/write operations creating new versions of the data. In bronze this is intended purely as a speed-up to reduce the amount of time spent iterating over the full bronze directory structure when later pipelines run.

The bronze pipelines have automatic alerts set up in the case of failures – these are currently configured to email a member of our team. The frequency/sensitivity of these alerts should be configured carefully based on how often the pipelines are running and the urgency of any potential fixes. We had periods where data collection was failing every run (due to external factors) which can lead to alert fatigue.

The demand forecast and actuals pipelines can handle many errors automatically – by saving any failed queries in a separate folder to be rerun 12 hours later. If collecting demand forecasts hourly, this may not be the best way to handle such cases. This functionality can also be used for 'backfilling' data after the fact – there were many occasions when we needed to pull large volumes of data after fixes were applied. The weather pipeline does not have this functionality. The API does not have the ability to backdate forecast requests so there is little benefit to pulling data in bulk. This is generally not a problem as the Met Office API is extremely reliable (we had only a single figure number of pipeline failures throughout the whole project duration). We are also rate-limited on the Met Office API (if this is exceeded then we lose access until the next day starts at midnight).

Azure Data Factory has limited ability to validate JSON schemas during ingestion (and even in our silver pipelines, tended to silently drop non-conforming data instead of alerting us). We therefore built an additional Python script to carry out separate schema validation. As well as ensuring that the data we received matched the format we required, this also whitelisted any text fields present to ensure that only the names of GSPs and DSRSPs were present, guarding against potential data breaches.

The use of JSON files as the smallest unit of data made tracking errors relatively straightforward – we were typically able to identify exact files responsible for validation failures. However, when there were issues affecting only a small number of data points (such as a single duplicated GSP/time combination from a flex event), the only way to correct this data was to replace the entire JSON file with a new version. If it is important for errors this small to be corrected during BAU, we recommend adjusting the pipelines to ensure that update/upsert options on the tabular data are available. This will mean single records can be corrected (and any duplicate data dropped if present) instead of needing to recollect entire files of mostly correct data to fix a single data point.

Silver

The silver data storage contains three main tables – one each for demand forecasts, actuals, and weather data. These are stored as 'Delta Lakes', a tabular format with built-in compression and scanning capabilities (and a more efficient storage structure than a human-readable format like csv). Delta Lakes also include look-back functionality, although we restricted some of this due to the frequency of read/write operations creating new versions of the data.

Gold

The gold inference table contains data merged from the silver demand forecast and weather tables. This is also the stage where we split apart GSP groups into the constituent GSPs. As well as giving better granularity in model predictions, this also ensures that there are no issues caused by having different groupings of GSPs between different trials/DSRSPs. This is currently the only processing or modelling step that uses information derived from GSP participant counts. We also assign unique identifiers to each row of data to ensure that we don't have rows with identical demand and weather forecasts (this could be caused where DSRSPs have returned the same forecast multiple times when we were querying multiple times each day). The inference table retains only one year of data, since we anticipate that most users are interested in future (or very recent) predictions. In BAU, this could be reduced further to speed up the UI response times (extracting the required forecasts from one year of continuous data may be slow, depending on how the Delta Lake is configured). The full training data can still be used to analyse any historical period. While we needed to use the event schedule for our silver validation, the inference table should be free of any reference to flexibility events or event details (such as duration or flex direction). This is to ensure full compatibility into BAU – where events are being planned based on model predictions, rather than scheduled far in advance.

The two training tables then further merge this data with the corresponding set of actuals – we anticipate that these will always correspond to known flexibility events. The split into AFM and EDM is to speed up read operations, since the majority of EDM data is not needed by the AFM (where rows correspond to the shoulders outside of flexibility events). The gold pipelines can also be used to insert any additional derived columns. This can be used to save time in model training: where necessary quantities can be calculated in advance. In practice, we still did most of these steps inside the modelling pipeline, where it was easier to modify and iterate the design incrementally. By contrast, updating columns in the gold tables requires a lengthier deployment process (since both Azure Data Factory and model code need changing) and then also needs the entire set of tables to be reprocessed to modify the available data. Once model features are stable long-term, more operations could be added to the Data Factory pipelines if desired.

Appendix B: Model input data

Here, we collect for convenience the inputs taken by the AFM and EDM, both at the level of the model endpoint (request sent by UI) and what is used by the machine learning models themselves.

Table 14 details the set of data sent by the user interface to the model (also needed for a user to query the endpoint directly). Table 15 details the full set of data needed for the model to give a sensible prediction. Most inputs are “direct” – corresponding values are used by the internal GBT as features to calculate the predicted flexibility. These are the ones for which it makes sense to evaluate feature importance. There are also number of “indirect” inputs. The model needs these to correctly filter rows and derive features – such as by restricting to a single trial type or summing over a region.

Parameter	Description	AFM or EDM
Start time	Start time of planned flexibility event	Both
Duration	Duration of planned flexibility event	Both
Flex direction	Turn-up or turn-down	Both
Region	Which region to query: 'DFS', 'LCM' or 'GSP'	Both
GSP list	List of GSPs to filter down to (may be null); only compatible with 'GSP' region and not currently available to set in the UI	Both
DSRSP list	List of DSRSPs to include	EDM Only
Target flex	Total target flex per settlement period (per GSP in 'GSP' region); may be set to null (blank in UI) to query AFM as proxy for target flex	EDM Only
Username	Username of account submitting request (only for logging purposes)	Both
Time of request	Time request is made (only for logging purposes)	Both

Table 14: Summary of inputs used by the model endpoints.

Parameter	Description	Type of input	AFM or EDM
DSRSP IDs	List of DSRPSs	Indirect – applied during data preprocessing	EDM Only
GSP ID	GSP corresponding row is for	Direct, target encoded	Both
Flex direction	Turn-up or turn-down	Indirect – applied by model before prediction is made	Both
Region	Which region to query: 'DFS', 'LCM' or 'GSP'	Indirect – applied during data preprocessing	Both
Forecast lead time	Time between forecast generation and the time it is predicting for	Direct	Both
Forecast demand	Predicted energy usage for settlement period	Direct	Both
Hour	Hour of day (0-23)	Direct, derived in feature extraction and target encoded	Both
Day	Day of week (1-7)	Direct, derived in feature extraction and target encoded	Both
Month	Month of year (1-12)	Direct, derived in feature extraction and target encoded	Both
Temperature	Average of max and min temperature	Direct, derived in feature extraction	Both
Wind speed	Average wind speed	Direct	Both
Rainfall	Total rainfall	Direct	Both
Humidity	Relative humidity	Direct	Both
Weather lead time	Time between forecast generation and the time it is predicting for	Direct	Both

Trial type	Availability or utilisation	Indirect – applied during data preprocessing	Both
Meter type	'Assets' or 'meters' (MPAN data)	Indirect – applied during data preprocessing	Both
Duration	Duration of flexibility event	Direct	Both
Target flexibility	Total target flexibility per settlement period	Direct	EDM Only
Time since event start	Time lapsed since start of flexibility event	Direct, derived in feature extraction	Both
Time through shoulder	Time before start or after end of event	Direct, derived in feature extraction	EDM Only

Table 15: Inputs used by AFM and EDM. 'Direct' means that they are used as predictors/features by the GBTs. 'Indirect' means they are used by the models in some other way, such as through an applied data filter or by changing the model internal state. Features may be taken from user input or from the gold data tables and may have additional processing or encoding applied. This table is for illustrative purposes only – for full technical specifications, please see the documentation accompanying the model code.

Appendix C: Model implementation details

As part of our work on model refinement, we compared the performance of two different model architectures.

We implemented gradient-boosted tree (GBT) models using the LightGBM and XGBoost libraries. LightGBM grows trees by focusing on developing the best leaf (per Features — LightGBM 4.6.0 documentation) whereas XGBoost models are grown a level at a time (not stated in their documentation). LightGBM may overfit on small datasets but tends to be faster. In order to reduce overfitting, the num_estimators and max_depth parameters can be used to restrict the number of trees and number of layers in each tree respectively.

We implemented the linear quantile regression (LQR) models using the scikit-learn library, augmented to use the commercial solver Gurobi to speed up fitting of the models. The coefficients for the LQR models are fitted within scikit-learn using a linear program solved using open-source solvers, of which HiGHs is the most performant. This fitting proved computationally intensive, so we implemented a Gurobi-based version to speed up fitting. Even with this augmentation, LQR models took far longer to fit than GBT models.

For the linear quantile regressor and LightGBM models, each quantile is represented by a different model while XGBoost internally handles quantiles, although still fits independent models for each quantile[26].

Throughout the user-facing parts of this project (UI, model endpoint including JSON export and this report), higher quantiles correspond to more flex in the requested direction. However, the code internally uses the common mathematical convention of higher quantiles corresponding to more positive flex. These conventions match for turn-up but are inverses for turn-down. We invert the quantiles in the function 'format_model_output' as part of converting the model predictions into a JSON export.

In order to tune the hyperparameters of the tree-based models, we performed a grid search over the following parameter ranges:

- learning_rate = [0.02, 0.04, 0.06, 0.08, 0.1, 0.12, 0.14, 0.16, 0.18, 0.2]
- max_depth = [3, 4, 5, 6, 7, 8, 9]
- n_estimators = [100, 90, 80, 70, 60, 50, 40, 30, 25, 10, 5]

To perform this grid search, models were trained with each combination of these parameters using the Ohme Availability data. Each model then has associated metrics: scaled quantile loss and coverage. In order to select the best set of hyperparameters, the median and 95% quantile were prioritised for turn-down models, and the median and 5% quantile were prioritised for turn-up models and a pareto-optimal selection was made to find the model with the best combination of performance metrics.

Code structure

The AFM and EDM each contain several tree models, one for each region and flex direction. However, MLflow requires a single entry point (fit and predict functions). We implemented a top-level model wrapper as this entry point, with minor specialisms between the AFM and EDM. Within the model wrapper is a hierarchy of classes to select the required tree model:

1. The class 'CFRegressorRegion' sits directly below the top-level wrapper and contains a sub-model for each of the three regions. For the 'DFS' and 'LCM' regional models, it handles the aggregation from per-GSP data (demand forecasts, weather forecasts etc.) to single values representative of the region.
2. Each regional model is an instance of 'CFRegressorFlexType', which contains two sub-models: one for each flex direction.
3. Now at the level of individual tree (or linear) models, each one is an instance of 'CFRegressor'.

[26] https://xgboost.readthedocs.io/en/latest/python/examples/quantile_regression.html#sphx-glr-python-examples-quantile-regression-py

Each of these classes has specialist implementations for each model type: LightGBM, XGBoost and linear. For example, initiating the top-level model wrapper with 'CFLGBRegressorRegion' will create a hierarchy of models all using LightGBM.

Automated performance tracking

Azure Machine Learning (Azure ML) has facilities for performing scheduled jobs on a dedicated compute instance. We developed code to create a compute cluster with permissions to access the models and write evaluation scores to the Azure ML workspace. Currently, the code is not set to run, because we are not receiving new demand forecasts (as outside trials) and so there is no new data to evaluate. Once activated as part of the transition to BAU, it will evaluate the most recently uploaded AFM at the time that the performance tracking schedule was created and perform this evaluation weekly at 03:00 UTC on Tuesdays. The code can be readily extended to cover the EDM too. Further ahead, we envisage a pipeline in BAU where model refitting and redeployment can occur automatically when the evaluation metrics indicate significant model drift has occurred, and the next scheduled evaluation is on this new model version.