Forecasting and Optimizing a Microgrid for the IEEE-CIS Technical Challenge

Richard Bean
School of Information Technology and Electrical Engineering
University of Queensland
St Lucia, Australia
r.bean1@uq.edu.au

***Abstract*—This paper provides a description of the methodology used in the IEEE Computational Intelligence Society 3rd Technical Challenge, which involved solar and building load forecasting, and optimization of the associated microgrid at Monash University. It achieved first place in the forecasting task and second place in the optimization task. The evaluation phase of the forecasting challenge required forecasting the 15-minute solar generation and building load of six solar installations and six buildings in a microgrid system at Monash University for the month of November 2020. Historical energy data was available with daily weather data from the Australian Bureau of Meteorology (BOM) and hourly data from the European Centre for Medium-Range Weather Forecasts (ECMWF). A quantile regression forest approach was chosen, using the weather variables provided, plus temporal variables. Novel thresholding approaches were used to improve the quality of the input data. As the training and evaluation phase of the challenge occurred during COVID-19 lockdown and reopening, the building demand was subject to pandemic-related effects. The approach used for the optimization task is described, which required mixed-integer programming (MIP) and mixed-integer quadratic programming (MIQP).**

***Index Terms*—time series forecasting, solar forecasting, renewable energy, random forests, optimization.**

I. Introducing the forecasting and optimization problems

In 2021, the IEEE Computational Intelligence Society ran a “predict + optimize” competition from 1 July to 3 November online [1].

The geographical context of the competition was at Monash University in Melbourne, Australia. In this microgrid, the electricity demand at a set of six buildings was met by a set of six solar installations, while a set of batteries with differing capacities and efficiency rates may be charged or discharged to meet requirements. A related optimization problem in the competition concerned how the energy requirements could be met at lowest cost using the batteries and foreknowledge of electricity prices.

The competition ran in two phases. In Phase 1 from 1 July to 11 October, competitors were able to upload forecasts to a public “leaderboard” which would calculate the Mean Absolute Scaled Error (MASE) [2] of each the twelve time series for October 2020. The mean of these twelve MASE values would then be displayed on the leaderboard within a few minutes, and there was no limit on the number of entries competitors could try. At the end of Phase 1, the load and solar data of October 2020 (the twelve time series) was made public, and Phase 2 began.

From 13 October to 3 November, competitors could upload forecasts, but the leaderboard only provided an indication of whether forecasts (for November 2020) were better, worse, or the same as a reference forecast of all zero values. On 3 November the MASE and energy cost figures were released, while the energy data was released on 6 December.

Section II describes the data available in the competition: prediction and historical energy data, and reviews the variables used in other competitions and papers. Section III discusses the approach used the develop the forecast for Phases 1 and 2 of the competition.

Section IV explains the optimization approach chosen and Section VI concludes the paper.

II. Data

A. Weather data

Competitors in the challenge were permitted to use external data from two sources: the Australian Bureau of Meteorology (BOM) weather data available through Climate Data Online [3] and the European Centre for Medium-Range Weather Forecasts (ECMWF) [4], [5] data provided by OikoLab [6]. Thus, the competitors were allowed to use “perfect forecast” weather data. From the BOM data, we used only the “daily global solar exposure” data, although daily minimum and maximum temperatures and daily rainfall data were available. The European Centre model is known as ERA5 and provides hourly historical weather (reanalysis) data.

The BOM “daily global solar exposure” data is measured from midnight to midnight each day, and is the total solar energy for a day falling on a horizontal surface. For the stations used in this study, values ranged from 1.3 to 32.3 MJ per square metre in 2019 and 2020.

The BOM daily global solar exposure data was available at three nearby sites - Oakleigh, Olympic Park, and Moorabbin (BOM sites 86077, 86088 and 86338 respectively).

B. Building and solar data

1) Solar: The solar installations, named Solar0 to Solar5, appeared to be of sizes of approximately 8 to 50 kW. The data for Solar0 only began in April 2020, but the estimated capacity factors of each installation (using data up to and including October 2020) ranged from 15.9% (Solar5, max 40.4 kW) to 23.5% (Solar1, max 12.7 kW).

For Solar0, 86% of the time-series values are non-zero; for Solar5, this proportion is 30%; while for the others, the value ranges from 38% to 48%. It seemed that some kind of cleaning or thresholding approach could be used to improve performance for forecasting Solar0 and Solar5 values and this proved to be true.

We thresholded the data from Solar1, Solar2 and Solar3 time series to begin from 22 May 2020. All of Solar0, Solar4 and Solar5 time series were used in training, although only hours where at least one period of generation (of four) was greater than 0.05 kW were used for Solar0 and Solar5 training.

2) Buildings: The provided data for the buildings and solar installations began at various dates, with the earliest data available from "Building 3" on 1 March 2016. The data for some buildings was very spotty; for example, Building 4 had 46,733 values but 18,946 of the values were unavailable. The modal and median value for Building 4 (19,621 occurrences) was 1 kW and all the other values were 2, 3, 4 or 5 kW. A perfect approach for data such as Building 4 would forecast one of these discrete values as the error rate for every series was significant for the competition, regardless of size.

We omitted one day in October 2020 from the building training data due to a public holiday effect in the data (Friday before Grand Final). It was unclear whether the public holiday was having an effect on this day, while November 2020 (the evaluation month) had no significant public holidays.

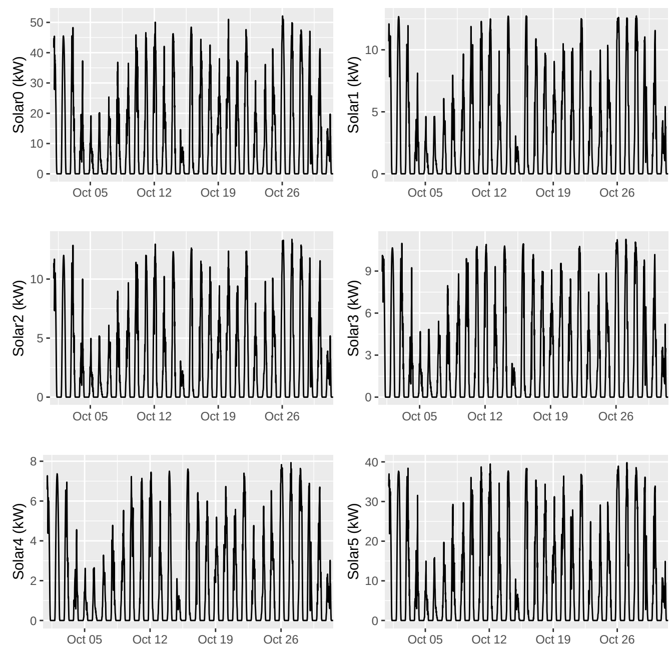

Fig. 1. Solar time-series data in kW for October 2020

III. Forecasting methods

A. Forecasting metric

The metric used in the competition, MASE, or Mean Absolute Scaled Error, is calculated as defined in [2], as follows.

Let Y_i, $i = 1, \ldots, n$ be the observations at time t and F_t be the forecast at time t, with the forecast error $e_t = Y_t - F_t$. Then calculate the scaled forecast errors q_t as

$$q_t = \frac{e_t}{\frac{1}{n-1}\sum_{i=2}^{n} |Y_i - Y_{i-1}|} \tag{1}$$

with MASE = mean($|q_t|$).

B. Initial Investigation

We began by using the Generalized Additive Model as seen in [7]. This was to develop an initial feel for how temperature and solar variables in the ECMWF (ERA5) data set affected each building and solar installation, along with temporal variables (weekend, time of day, and day of year).

We noticed that the buildings were very different in terms of load on weekend and public holidays (see Building 1, 3 and 6 in Figure 2), and that temperature and solar (leading and lagging) were the most critical predictor variables in the models for the buildings and solar output.

We quickly switched to a random forest model as the focus of the competition was purely the lowest error rate, rather than explainability or visualization.

The *ranger* package [8] in R has provided multi-threaded random forests with an extension of options over those seen in the original *quantregForest* package of [9].

A plot of the solar and building traces for October 2020 (the data held out for Phase 1) is shown in Figures 1 and 2.

The solar traces seemed to be genuine 15 minute readings, while Buildings 0 and 3 were series of 15 minute values repeated 4 times each. It seemed that Building 4 and Building 5 readings were uncorrelated or poorly correlated to any weather or temporal variables we were provided. Thus, for the November 2020 forecast, we simply repeated the median value from October 2020 (i.e. "manual optimisation"). This observation saved time in the prediction development and iterative process as the observations for only the other four, rather than all six, buildings were used in the combined training data.

We thresholded values from Building 0 and 3 as some of them appeared to be large outliers. Building 0 and 3 upper bounds were set to 606.5 and 2264 kW while the Building 3 lower bound was set to 193 kW.

A "maximal" approach was used; that is, for each building time series, the training data start date was decreased month by month as far as possible until the error rate started increasing. For the building time series, this was the months of June, February, May, and January 2020 for Buildings 0, 1, 3 and 6 respectively. That is, for Building 0, the training data for

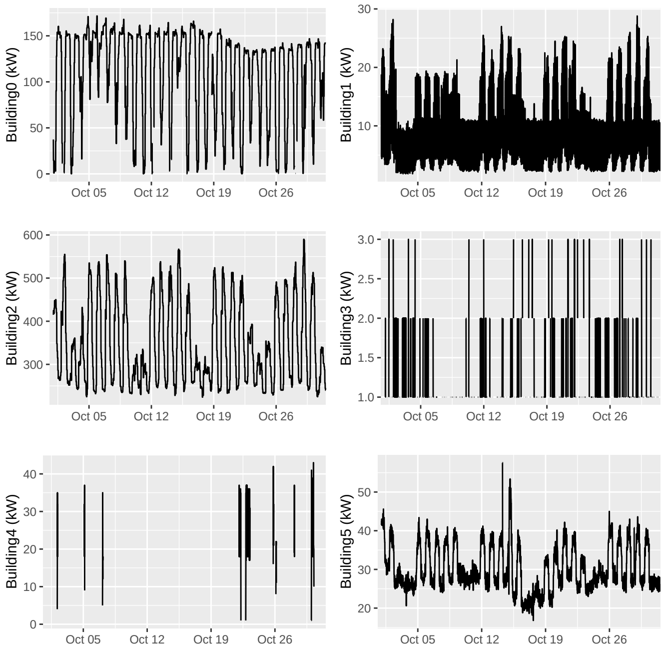

Fig. 2. Building time-series data in kW for October 2020

Phase 1 consisted of June to September 2020 inclusive, while for Phase 2 the training data was from June to October 2020 inclusive. It was assumed that all the most recent data should be included in training. We attempted to add a recency bias for newer data following [10] using exponential decay in the *ranger* training, but this was unsuccessful.

The approach chosen here was the most difficult and most purely subjective choice made by us in the competition. It was assumed that a full reopening after COVID-19 restrictions were lifted (in October 2020) would not result in a return to pre-2020 levels of building energy use. For example, the maximum Building 3 energy use observed was 683 kW after May 2020 but there were observations of over 1,000 kW on 19 March 2020. The last observation over 700 kW was on 27 March 2020 after which it was assumed stricter COVID-19 lockdowns commenced.

C. *Forecast code*

The forecast code was built in R due the availability of useful packages such as *ranger*, *xts*, and *lubridate*.

The following list shows the pseudo-code loop we used for the model development. The same R code was used for Phase 1 (October) and Phase 2 (November) with only the phase parameter changed (although different random seeds would perturb the output slightly).

- Pick a time series or group of time series
- Perform adjustment:
  - Adjust start and end dates of training data
  - Perform thresholding (effective for Solar 0 and Solar 5 series)
  - Add or subtract predictor variables (e.g. leading and lagging variables, BOM variables)
  - Group solar or buildings together differently
  - Adjust random forest parameters (number of trees and mtry)
- Assess MASE of Phase 1; if a change has resulted in a lower MASE, then retain the change, otherwise discard it

Feature selection was generally performed manually using our knowledge of other competitions and experience in solar and energy forecasting, rather than an automated step-wise process of addition and subtraction of variables. This was motivated by time pressure of the competition. Initially all groups of buildings and solar installations were trained together, and the highest variables in terms of importance across all of the group were extracted. The idea was to avoid overfitting and save time by choosing only directly relevant variables and testing only against a proxy for the evaluation metric - the MASE of Phase 1. In a production environment the validity of these assumptions should be verified through cross-validation.

After performing feature selection for each building and setting the value of Building 4 to be 1 kW for the whole month of October, we achieved an error rate (MASE) for Phase 1 of 0.6528. This required the selection of start months for the buildings and solar data, further refined in Phase 2.

By the end of Phase 1, we had lowered this to 0.6320 by incorporating median forecasting of the time series (that is, the 50th percentile in quantile regression) and adding in BOM solar data.

On 13 October, the individual Phase 1 time series became available so we investigated how the MASE value was derived using the provided data and R program.

We then added the following improvements sequentially through experimentation as each change seemed reasonable. We thought each change would improve the error rate for Phase 2 (November 2020) as well. The possibility of overfitting seemed minimal as each of the changes could be justified with reference to the closest month of October 2020.

- Added cloud cover variables $\pm$ 3 hours; MASE 0.6243, 16 October
- Selected solar data from beginning of 2020 instead of from day 142 (22 May); MASE 0.6063, 17 October
- Selected start month (0-8) for each of four building series from 2020, added all possible weather variables, set building 5 equal to median training value 19 kW; MASE 0.5685, 18 October
- Fixed up Solar5 data by filtering out values less than 0.05 kW; MASE 0.5387, 24 October
- Observed that forecasting Solar0 and Solar5 as linear combinations of the other Solar variables was working better than the actual Solar0/5 prediction
- Observed that some pairs of solar series were much more highly correlated than other pairs, and buildings 3/6 were also highly correlated
- Tested exponential decay ideas, which were unsuccessful

- Trained all solar and building data together following [11] (MASE 0.5220, 30 October).
- Fixed up Solar0 data by same filtering as for Solar5 (MASE 0.5207, 31 October)
- Added in separate binary variables for each day of the week (MASE 0.5166, 2 November)

Based on this result, we were expecting a similar MASE for Phase 2; however, the "reopening" effects of lockdown resulted in a reversion to historic usage patterns in some of the buildings, which diverged from our forecast.

Naturally this "reopening" affected all competitors. In a production environment, instead of having a "month-ahead" forecast, the forecasts would be day-ahead and able to rapidly adjust to reopening effects.

Although we do not know the types of the buildings, we surmise that in November 2020 the air-conditioning use of some of them began to revert to the long-term mean. Thus, our approach of choosing different starting months for each building to minimize MASE vis-a-vis October 2020 led to model outperformance.

D. Model description

We summarise the predictor variables used in the building and solar modelling and discuss their relative importance.

The building predictor variables are being used to predict the quarter-hour energy usage for each of the buildings; that is, a different model is used for each quarter-hour offset (:00, :15, :30 and :45 of each hour). In the final model, all the solar and building variables (for Buildings 0, 1, 3 and 6) were normalized using the maximum value found in the training data.

The weather variables (t2m, d2m, wind, MSLP, R, SSRD, STRD and TCC) are the variables provided by OikoLab via the ERA5 model: 2 metre temperature, 2 metre dewpoint temperature, wind speed, mean sea level pressure, relative humidity, surface solar radiation downward, and surface thermal radiation downward.

We added an indicator variable to identify the building being predicted, analogously to the same variable in [11]. Each weather variable had leading and lagging variables added one, two or three hours from the period. Lagged temperatures of 24, 72 and 48 hours before are used for building training, based on the market demand modelling of [12]. These proved to be significant variables in the building energy forecasting.

The variables Moorabbin, Oakleigh and Olympic are repeated values for the BOM daily solar global exposure variables at three sites. That is, in each quarter hour, the variable is assigned the daily solar global exposure for that day, as the value is measured from midnight to midnight.

The variables "sin_hr" and "sin_day" refer to the Fourier terms related to the hour of the day and the day of the year (Julian date). Thus $sin\_hr = sin(\frac{2\pi hr}{24})$ and $sin\_day = sin(\frac{2\pi day}{365})$ and similarly for the cosine term. These terms model the diurnal and annual cycle in the building energy usage and solar generation. Including these temporal terms, plus the weekday/weekend Boolean variable for the buildings, gives a good first approximation to the building energy usage and solar generation.

These terms are also seen in other competition winning entries such as [13] where the competitors used these terms in both the solar and wind forecasting tracks of the 2014 Global Energy Forecasting competition.

Others variables are binary for weekend ("wd"), Monday/Friday ("wd1"), Tuesday/Wednesday/Thursday ("wd2") and named variables for each day of the week.

Out of the individuals/teams who made submissions to the evaluation Phase, our entry had the lowest MASE for Phase 2 of 0.6460. From six entries with known time series MASE, it achieved the lowest MASE on three buildings, the equal lowest MASE on one other building, and the lowest MASE on each of the six solar installations.

The outperformance is due to many factors including the fine-tuning described above in Subsection III-C. We believe that relative to other competitors the approaches of thresholding each building input data set differently, modelling all solar time series together, and including both daily and hourly weather data in the model led to its strong outperformance.

IV. Optimization

The optimization task was to minimize a cost which included three terms: a quadratic term proportional to the monthly peak load, a linear term reflecting the cost of energy used during the month, and costs (or benefits) related to the scheduling of *once-off* activities.

The constraints related to the scheduling of *recurring* or *once-off* activities. Each activity required the use of a specified number of large or small rooms in each building, used a given number of 15 minute periods, and a specified amount of energy. Each building has a given number of large or small rooms. The cost was to be minimized over ten provided instances: five *small* each with 50 recurring/20 once-off activities and five *large* each with 200 recurring/100 once-off activities.

The recurring activities had to be scheduled within office hours (9am-5pm weekdays) meeting a series of precedence constraints based on the day of the week, and these recurred in every week of the month scheduled. Once-off activities could be optionally scheduled, and would receive a bonus if scheduled, and a penalty if scheduled outside of office hours. Such activities had to meet constraints based on the day of the month.

In addition, two batteries were present, which could be charged or discharged in any period, and had different capacities and charge/discharge rates. These could be used to either reduce the effect of high pool-price periods, or decrease the peak load over the month, or both. Further details may be found at [1].

Here, O represents the objective function to be minimized, l_t is the net load (buildings minus solar) in period t, e_t is the wholesale energy cost in period t, d_i is a Boolean variable reflecting whether "once-off" activity a_i is scheduled, o_i is a Boolean variable reflecting whether the activity is scheduled in office hours, while $value_i$ and $penalty_i$ reflect the benefit or cost, respectively, of scheduling the once-off activity.

$$O = \sum_t \frac{0.25 l_t e_t}{1000} + 0.005(\max_t l_t)^2 - \sum_{a_i}(d_i(value_i - o_i penalty_i)) \quad (2)$$

A four-step approach was used using the forecast task output.

First, a mixed-integer program (MIP) was solved for the recurring and recurring plus once-off activities, then each of these was extended using a mixed-integer quadratic program (MIQP).

The general strategy was chosen from one of two ("array" from the "array" and "tuples" approaches) while the specific step improvement strategy was chosen from one of five ("no forced discharge").

The optimization code was written in Python using Gurobi as a solver. Thus, the key input files for the Python optimization were the vectors of prices for the month plus the vector of "net" load (that is, sum of building load minus sum of solar generation) for the month; 2,976 values for October 2020 (Phase 1) and 2,880 values for November 2020 (Phase 2).

After examining the Phase 2 instances, we decided to try to include all the once-off activities only in "peak" periods as this was much easier in MIP terms and the sum of the penalties for scheduling the once-off activities in "off-peak" exceeded the benefits for scheduling them in "peak" in every instance.

- Develop a MIP for each of 5 small and 5 large instances - minimize the recurring load over all peak time periods
- Extend the solution of each MIP (Phase 2) to include all once-off activities in peak
- For each of 10 instances, solve the MIQP: attempt to add batteries and shift activities using the "no forced discharge" approach described below; consider all intermediate solutions
- Perform the same task for the "recurring plus once-off" solutions found
- Assess the cost using the objective function; if the "recurring plus once-off" cost for an instance is lower than the "recurring only" cost choose that solution, otherwise choose the "recurring only" solution

The approach is shown in Figure 3.

V. Experiments

After attempting to solve the 'small' and 'large' instances as mixed-integer quadratic programs (MIQPs) with the quadratic term in the objective function, we soon realized that the cost of electricity from the pool prices (wholesale price) varied little between solutions, and the best solution would be much more easily found by solving a MIP (mixed integer program).

That is, first minimize the recurring load over all peak time periods for the set of small and large instances, store all the intermediate solutions, and then attempt to solve the MIQP (incorporating the peak quadratic term in the objective function) by allowing Gurobi to check if moving activities around from the intermediate solutions could decrease the cost. Attempts to add constraints to the MIP to bias the solution away from weekdays with a higher average price were unsuccessful.

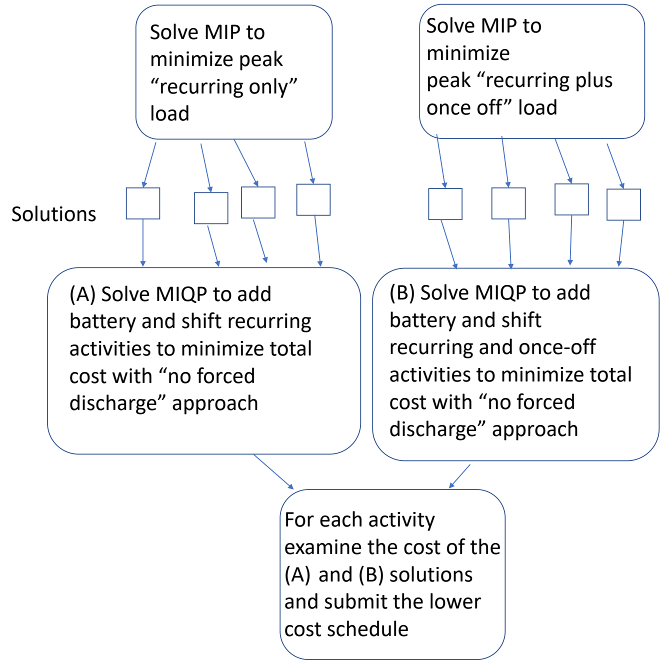

Fig. 3. Phase 2 Optimization Solution Approach

The organizers envisaged that forecast skill would have more effect in Phase 2 than Phase 1, but judging by the final Phase 2 leaderboard, it seems that there was little correlation. Perhaps having a commercial solver such as Gurobi and access to high-performance computing facilities were more important factors. In contrast, the forecasting task could be performed on a single computer in minutes.

A paper [14] examined developing scheduling algorithms for home battery/inverter combinations.

In that work, a key design decision was never to charge in peak hours and assess cost of different battery scheduling approaches over 83 inverters. The approaches included PV persistence, PV and Load persistence, Load persistence, quantiles of 50/50 and 60/40 for the PV and Load, and persistence of the last hour.

In the current work, we considered five approaches:

- *Conservative* is just choosing the lowest recurring load and lowest recurring + once off load and evaluating cost using a naive or flat forecast. This was probably the winning approach for cost in Phase 1, as some competitors had winning results with no forecast, or a poor forecast, but seemed pointless to me as the organizers said quality of forecast should contribute to results in phase 2.
- *Forced discharge* forbids any charging in peak hours, and forces at least one of the two batteries to be discharging in every peak period. This was thought to avoid nasty

surprises in the peak load as in phase 1 one of the actual observed values (period 2702 of 2976) was 260 kW above my final forecast (i.e. forecast with 0.5166 MASE). However, although values drop randomly in and out of the building data, we hoped that there were no "outliers" in phase 2 as promised (although this "outlier" comment from the competition organizers probably referred to the repeated 1744.1 kW values in the Building 0 trace - periods 1710 to 1713 of 2976).

- *No forced discharge* forbids any charging in peak hours, but the MIQP solver decides whether to discharge or do nothing in those hours.
- *Liberal* allows charging in peak, but the maximum of recurring + once off + charge effect for each period is limited to the maximum of recurring + once off load over all periods. This is to avoid nasty surprises when the solver thinks that a period has low underlying load and schedules a charge (due to a low price in that period) but then accidentally increases the maximum load over all periods, which can be very costly.
- *Very liberal* allows charging over peak and does not attempt to control the maximum of recurring + once off + charge effect. This would be the best approach if the forecast was perfect.

Each approach was assessed starting with the best Phase 1 solutions obtained (and the best forecast available for Phase 1). A Java program provided by the competition organizers was used to calculate the cost of each approach.

It was found the "liberal" and "very liberal" approaches resulted in the lowest objective function value for the MIQP; but the prices obtained were actually higher using the known load and solar values. Over the 10 sample problems, the total cost was lowest for "no forced discharge" (evaluated prices: $396,264 for Forced, $396,060 for No Forced Discharge) and so this approach was used for Phase 2. Ultimately only "Large 2" and "Large 4" solutions included once-off activities.

The estimated final cost for Phase 2 was $261,906 and after the final leaderboard was published, the actual cost was $335,107. This underestimate was due to increases in energy use at several of the buildings.

VI. Conclusion

In order to forecast 15-minute energy and solar time-series data, we applied the quantile regression forest of [9] based on the original random forest idea of [15] as provided in the R *ranger* package [8]. This was highly effective in conjunction with techniques of thresholding, grouping related buildings and solar installations, combining daily and hourly data from two uncorrelated data sources, and normalization. The quantile regression forest approach was ideal as it required minimal parameter tuning and thus the process of building and testing models was expedited in the time-pressured environment of the competition.

For each phase, the training data for each building and solar installation was extended back from the latest available data, month by month, until the error rate for each building or solar time series began to increase. In a production environment as opposed to a competition, this approach should be verified by performing cross-validation. This approach of grouping and thresholding may have captured different building types present on campus and their differing response to reopening, as seen in studies of the effect of Covid-19 lockdowns on energy at universities campuses worldwide.

The success of combining daily solar exposure data with hourly surface solar radiation data was unexpected and played a large part in the outperformance of the approach versus the other contest entrants.

For optimization, we used a two-step process of building a mixed-integer program and extending this to a mixed-integer quadratic program to minimize the cost over a month. After examining five approaches, we chose an approach of *no forced discharge* of batteries to extend each MIP.

The code used may be found at online at [16].

Acknowledgment

This work was partially supported by funding from Project SHIELD of the Australian Renewable Energy Agency.

References

[1] C. Bergmeir, F. de Nijs *et al.*, "IEEE-CIS Technical Challenge on Predict+ Optimize for renewable energy scheduling," 2021. [Online]. Available: https://dx.doi.org/10.21227/1x9c-0161

[2] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting*, vol. 22, no. 4, pp. 679–688, 2006.

[3] Bureau of Meteorology, "Climate Data Online," 2022. [Online]. Available: http://www.bom.gov.au/climate/data/index.shtml

[4] H. Hersbach and et al, "ERA5 hourly data on single levels from 1979 to present," 2018. [Online]. Available: https://doi.org/10.24381/cds.adbb2d47

[5] ——, "ERA5 hourly data on pressure levels from 1979 to present," 2018. [Online]. Available: https://doi.org/10.24381/cds.bd0915c6

[6] J. Yang, "OikoLab," 2022. [Online]. Available: https://www.oikolab.com

[7] A. Pierrot and Y. Goude, "Short-term electricity load forecasting with generalized additive models," *Proceedings of ISAP power*, vol. 2011, 2011.

[8] M. N. Wright and A. Ziegler, "ranger: A fast implementation of random forests for high dimensional data in C++ and R," *arXiv preprint arXiv:1508.04409*, 2015.

[9] N. Meinshausen and G. Ridgeway, "Quantile regression forests," *Journal of Machine Learning Research*, vol. 7, no. 6, 2006.

[10] P. Wang, B. Liu, and T. Hong, "Electric load forecasting with recency effect: A big data approach," *International Journal of Forecasting*, vol. 32, no. 3, pp. 585–597, 2016.

[11] S. Smyl and N. G. Hua, "Machine learning methods for GEFCom2017 probabilistic load forecasting," *International Journal of Forecasting*, vol. 35, no. 4, pp. 1424–1431, 2019.

[12] A. E. Clements, A. Hurn, and Z. Li, "Forecasting day-ahead electricity load using a multiple equation time series approach," *European Journal of Operational Research*, vol. 251, no. 2, pp. 522–530, 2016.

[13] R. Juban, H. Ohlsson, M. Maasoumy, L. Poirier, and J. Z. Kolter, "A multiple quantile regression approach to the wind, solar, and price tracks of GEFCom2014," *International Journal of Forecasting*, vol. 32, no. 3, pp. 1094–1102, 2016.

[14] R. Bean and H. Khan, "Using solar and load predictions in battery scheduling at the residential level," *arXiv preprint arXiv:1810.11178*, 2018.

[15] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[16] R. Bean, "IEEE Predict Optimize Challenge," 2021. [Online]. Available: https://github.com/RichardBean/IEEE-Predict-Optimize-Challenge