Battery Scheduling Algorithms for Peak Demand Management and Demand Response

Shan Dora He
dora.he3@monash.edu

December 15, 2021

Executive Summary

Large electricity consumers pay a *network peak demand charge* based on their maximum demands in addition to their actual electrical consumption per day. They can also participate in *demand response* to receive financial rewards by reducing consumption at times with high electricity wholesale prices. One way to reduce network peak demand charges and earn rewards without changing existing consumption patterns is to use batteries. We can charge batteries during times with low demands or prices and discharge them at times with high demands or prices.

This work develops a scheduling algorithm that decides the best times to charge and discharge batteries, in order to minimise the charges and maximise the rewards. This algorithm combines linear programming and a rolling horizon rescheduling strategy to schedule batteries based on price and demand forecasts and incorporate uncertainty in these forecasts. The results show that we can reduce the annual network peak demand charge of Monash Clayton campus by 2% at most and the monthly network peak demand charge by 13% at most for 2020 using the existing Vanadium Flow battery and Lithium-ion battery. The results for demand response are still under investigation.

In order to hedge against the risks of uncertain future and inaccurate forecasts, future work can consider employing additional methods to incorporate various future scenarios, and scheduling batteries based on the expected outcomes of those future scenarios instead of a single forecast. Other future work can include investigating the scheduling results using different sizes of batteries, and evaluating the economic costs and returns of those battery sizes.

List of Acronyms

battery battery energy storage system.

DR demand response.

LP linear programming.

NPDC network peak demand charge.

PDM peak demand management.

RHRS rolling horizon rescheduling strategy.

SOC state-of-charge.

List of Symbols

B^{bat} the total number of batteries.

$b^+_{i,n}$ the amount of power charged to the battery at time step n.

$b^-_{i,n}$ the amount of power discharged from the battery at time step n.

e^{init}_i the initial battery energy level.

e^{max}_i the maximum battery capacity.

$\bar{p}_i$ the maximum battery power rate.

e^{min}_i the minimum battery capacity.

η_b the battery round-trip efficiency.

$soc_{i,n}$ the state of charge at time step n.

$\hat{d}^{ann}$ the pre-defined maximum demand threshold in the hours of the annual peak demand tariff.

$\bar{d}^{ann}$ the maximum demand in the hours of the annual peak demand tariff.

.

l^{fore}_n the forecast load at time step n in MWh.

$\hat{d}^{summ}$ the pre-defined maximum demand threshold in the hours of the summer monthly peak demand tariff.

$\bar{d}^{summ}$ the maximum demand in the hours of the summer monthly peak demand tariff.

f^{cost} the total peak demand charge.

$g^{penalty}$ the penalty cost of charging/discharging batteries frequently.

$h^{combined}$ the combined objective.

w^{cost} the objective weight of the peak demand charge.

$w^{penalty}$ the objective weight of the penalty costy of charge/discharge batteries frequently.

$\mathbf{t}^{ann}$ binary indicators of hours that the annual maximum demand tariff is applied to.

p^{ann} the annual maximum demand tariff.

p_n^{spot} the wholesale spot price for the trading interval n in \$ / MWh.

$\mathbf{t}^{summ}$ binary indicators of hours that the summer monthly maximum demand tariff is applied to.

p^{summ} the summer monthly maximum demand tariff.

N^{day} the total number of time steps per day.

N^{hour} the total number of time steps per hour.

T total minutes per time step.

Contents

Chapter 1

Introduction

1.1 Project Goals

Large electricity consumers, such as a customer, a company or a premier with 400 MWh electrical consumption per year, pay a *network peak demand charge (NPDC)* in addition to their actual electrical consumption per day. These consumers also can receive financial rewards for reducing loads when the wholesale price is above a threshold via contracts with the retailer.

The goal of this project is to reduce the NPDC and increase rewards by modifying existing demands using battery energy storage systems (batteries). To achieve this goal, this project develops computer algorithms that:

- automatically calculate the best times to charge or discharge batteries based on price and demand forecasts,
- minimises the peak demand charges,
- maximises the financial rewards.

This report presents the development of models and algorithms for peak demand management (PDM) and wholesale market demand response (DR) as part of the Smart Energy City project of Monash University on the Clayton campus.

1.2 Methodology of Battery Scheduling

Optimisation is a field that studies methods for finding best decisions for problems that minimise or maximise objectives of these problem. Constraints are applied to decisions, so that a finite number of options are available for each decision. In the context of battery scheduling, optimisation can refer to a method that decides the best amount of energy to be charged or discharged at each time interval of the day, in order to minimise costs and maximum profits. The amount of energy to be charged or discharged is restricted by the capacity and power rate of the battery, and the demand of the consumer.

Linear programming (LP) is a type of optimisation methods that have been widely used for solving battery scheduling problems in the literature [1, 2, 3, 4, 5]. However, it is insufficient to simply apply optimisation methods to schedule batteries once per day based on forecasts, because of the presence of uncertainty in future demand [6, 7, 8]. One way to incorporate uncertainty is to use a *rolling horizon rescheduling strategy (RHRS)*, which is sometimes also known as *rolling horizon control strategy* or *rolling horizon predictive strategy* in the literature [6, 7, 5, 9].

Typically, a RHRS involves forecasting demand and prices, and rescheduling batteries iteratively during a day. At each iteration, the future demands and prices are re-forecasted using updated information, and battery schedules are re-computed given the new forecasts. The battery schedule from the last iteration will be discarded and the new battery schedule will be implemented until the next iteration where new demand forecasts and battery schedules are calculated again. More details of this RHRS are explained in [6] and Section 2.5.

This work combines LP methods and the RHRS to schedule batteries in the context of the Monash smart city project, in order to minimise the NPDC and maximum rewards from DR. This document presents the formulation of the battery scheduling problems, the development of solving methods and the evaluation of these methods for both peak demand management and DR.

The detailed implementation of this work is available on BitBucket `https://bitbucket.org/dorahee2/battery-scheduling/src/master/`. At the current stage of the project, this repository needs to remain private. Please email `dora.he3@monash.edu` for access to this repository.

1.3 Report Structure

This report is structured in the following way:

- Chapter 2 presents the model and development of the battery scheduling method for peak demand management.
- Chapter 3 presents the model and development of the battery scheduling method for demand response.

Chapter 2

Peak Demand Management

2.1 Summary

Large electricity consumers pay a *network peak demand charge* in addition to their actual electrical consumption per day. Reducing peak demand can yield significant savings for Monash University. One way to reduce peak demand without changing existing consumption patterns is to use batteries. We can minimise the peak demand and therefore the network peak demand charge by fully charging batteries before peak times and fully discharging them at peak times. To achieve this goal, this work develops a method that combines linear programming and rolling horizon rescheduling strategy to automatically schedule batteries based on demand forecasts. The results show that by using the existing Vanadium Flow battery and Lithium-ion battery, we can reduce the annual network peak demand charge of Monash Clayton campus by 2% at most and the monthly network peak demand charge by 13% at most for 2020.

2.2 Peak Demand Tariff

Large electricity consumers, such as a customer, a company or a premier with 400 MWh electrical consumption per year, pay a *network peak demand charge (NPDC)* in addition to their actual electrical consumption per day. For the Clayton campus of Monash University, this NPDC includes two (2) components:

- An annual maximum demand component: This component is calculated based on the maximum average demand in 30 minutes on a rolling 12-month basis. This rolling 12-month period is defined as the 11 full calendar months prior to the current day and the current month. Particularly, this component is applied to 30-minute intervals from 7am to 7pm on Monash work days. The tariff for this component is \$131.7 / MVA per day, which is about \$48k per MVA per year. The definition of Monash work days is provided in Section 2.4.1.

- A monthly maximum demand component for summer months: This component is calculated based on the maximum average demand in 30 minutes for each full calendar month over the five summer periods from 1 Nov to 31 Mar. Particularly, this component is applied to 30-minute intervals from 3pm to 6pm on Monash work days during the five summer months. The tariff for this component is \$162.5 / MVA per day, which is in total \$24.5k per MVA per five-summer-month.

The tariffs for the NPDC vary annually. However, in recent years, the changes are within plus or minus 5—10% of the above numbers. Reducing peak demand can yield significant savings for Monash University. Imagine a scenario where 1 MVA is reduced from the highest average demand in a 30-minute interval, it will yield approximately \$72000 or up to \$ 144 / MWh, which is roughly ten times of the maximum wholesale price cap for electrical consumption, or a thousand times of retail electricity rates.

2.3 Methodology of Battery Scheduling for Peak Demand Management

One way to reduce peak demand effectively without changing existing consumption patterns is to use battery energy storage systems (batteries). We can minimise the NPDC by fulling charging batteries during off-peak times and discharging them at peak times. The minimisation of the NPDC using batteries can be achieved by computer algorithms that predict future demands based on historic demands, and plan battery operations in advance in order to charge and discharge batteries at the best times. This work is interested in such algorithms. This work follows the methodology explained in Section 1.2 to optimally schedule batteries and incorporate uncertainty in demand forecasts, using *linear programming (LP)* models and the *rolling horizon rescheduling strategy (RHRS)*.

2.4 Battery Scheduling Problem for Peak Demand Management

This work studies a battery scheduling problem where batteries are scheduled to minimise the network peak demand charge (NPDC) given the annual peak demand tariff, the monthly peak demand tariff for summer months, demand forecasts and historic demands. This section presents the input parameters, variables, constraints, objective functions and mathematical formulation of the battery scheduling problem.

2.4.1 Problem Parameters

The input parameters for this battery scheduling problem include time intervals, battery specifications, peak demand tariffs and applicable hours, pre-defined demand thresholds, demand forecasts and historic demands.

Time Interval

A day is considered as a finite number of time intervals for scheduling. We assume that the charge or discharge rate of a battery does not change in a time interval. Let us write:

- T as the length in minutes of each time interval,
- $N^{day} = 1440/T$ as the total number of time intervals per day,
- $N^{hour} = 60/T$ as the total number of time intervals per hour,

We have chosen $T = 30$ mins for this work, because at the time of this work, the wholesale electricity prices were updated every thirty minutes.

Battery Specification

The total number of batteries is denoted as B^{bat}. Each battery $i \in [1, B^{bat}]$ is modelled by the following elements:

- e_i^{init}: the initial energy level at the beginning of the day (in kWh)
- e_i^{min}: the minimum allowed energy capacity (in kWh)
- e_i^{max}: the maximum allowed energy capacity (in kWh)
- $\bar{p}_i$: the maximum power rate (in kW)
- $b_{i,n}^{+}$: the amount of power charged to the battery at time interval n (in kW):
- $b_{i,n}^{-}$: the amount of power discharged from the battery at time interval n (in kW)
- η_b: the round-trip efficiency (between 0 and 1)
- $soc_{i,n}$: a state-of-charge (SOC) profile — the amount of energy remaining in the battery at time interval n (in kWh)

Peak Demand Tariff and Applicable Hour

The peak demand tariffs include two components:

- The annual maximum demand component: p^{ann} = \$131.7 / MVA.
- The summer monthly component: p^{summ} = \$ 162.5 / MVA.

Although the NPDC is calculated in MVA, in the Monash context, MW can be used as proxy, due to persistent near unity Power Factor (typically at 0.98 or 0.99). Furthermore, the \$ / MVA based charge applies to 1/2 hour with the highest MW's. For example, the selection criteria for the peak is unchanged between MVA, on the basis of the charge (\$ / MVA) differs by 1 - 2 %. 1 MVA = 1 MW / Power Factor.

Note that both tariffs are applied to Monash work days, which are defined in `https://www.monash.edu/students/admin/dates/holidays`. We have also introduced binary variables to indicate the hours that a tariff is applied to:

- The hours to which the annual maximum demand tariff is applied: $\mathbf{t}^{ann} = \{t_n^{ann} = 1 \ if \ 7 \times 2 - 1 \leq n \leq 19 \times 2 \ else \ 0 \mid n \in [1, N^{day}]\}$.
- The hours to which the monthly maximum demand tariff is applied: $\mathbf{t}^{summ} = \{t_n^{summ} = 1 \ if \ 15 \times 2 - 1 \leq n \leq 18 \times 2 \ else \ 0 \mid n \in [1, N^{day}]\}$.

We have also considered a pre-defined maximum demand threshold for each tariff:

- The pre-defined maximum demand threshold for the annual maximum demand tariff: $\hat{d}^{ann}$.
- The pre-defined maximum demand threshold for the monthly maximum demand tariff: $\hat{d}^{summ}$.

These two pre-defined maximum demand thresholds can be given by users or calculated from historic demands.

Demand Forecast

Demand forecasts are required to plan the battery operations in advance. At the time of this work, at each time interval n, the demand is forecasted from the upcoming time interval $n + 1$ until 6pm of the same day. However, to allow batteries to charge after 6pm, we assume the forecasted demand after 6pm is zero. Let us write d_n^{fore} as the demand per time interval in kW.

2.4.2 Historic Demand

Historic demands are used to calculated the maximum demands in order to calculate the NPDC. Ideally, at each time interval n, the historic demands should be for the 12 months prior to the current time interval. If less than 12 months of data is provided, the maximum demands for peak demand tariffs will be calculated from the available data. Let us write the historic demands at time interval n as d_n^{his}.

2.4.3 Problem Variables

The decision variables include the amount of power charged to or discharged from each battery at each time interval: $b_{i,n}^+$ and $b_{i,n}^-$. Other variables include:

- The modified demand per time interval d_n^{mod}, calculated as Equation 2.1.

$$\forall n \in [1, N^{day}],\ d_n^{mod} = d_n^{fore} + \sum_{i=1}^{B^{bat}} (b_{i,n}^+ + b_{i,n}^-) \tag{2.1}$$

- The maximum demand in the hours that the annual peak demand tariff is applied to $\bar{d}^{ann}$, computed as Equation 2.2 and 2.3.

$$\forall n \in [1, N^{day}],\ \bar{d}^{ann} \geq d_n^{mod} \times t_n^{ann} \tag{2.2}$$

$$\bar{d}^{ann} \geq \hat{d}^{ann} \tag{2.3}$$

- The maximum demand in the hours that the summer monthly tariff is applied to $\bar{d}^{summ}$, calculated as Equation 2.4 and Equation 2.5.

$$\forall n \in [1, N^{day}],\ \bar{d}^{summ} \geq d_n^{mod} \times t_n^{summ} \tag{2.4}$$

$$\bar{d}^{summ} \geq \hat{d}^{summ} \tag{2.5}$$

2.4.4 Problem Constraints

Each battery b has the following constraints:

- *Charge or discharge only constraint*: At each time interval n, each battery can either charge or discharge, described as Equation 2.6.

$$\forall i \in [1, B^{bat}],\ \forall n \in [1, N^{day}],\ b_{i,n}^+ \times b_{i,n}^- = 0 \tag{2.6}$$

- *Maximum power constraint*: At each time interval n, each battery cannot charge or discharge at a rate higher than the maximum power rate, defined as Equation 2.7 and 2.8.

$$\forall i \in [1, B^{bat}],\ \forall n \in [1, N^{day}],\ 0 \leq b_{i,n}^{+} \leq \bar{p}_i \tag{2.7}$$

$$\forall i \in [1, B^{bat}],\ \forall n \in [1, N^{day}],\ 0 \leq -b_{i,n}^{-} \leq \bar{p}_i \tag{2.8}$$

- *Maximum-minimum capacity constraint*: At each time interval n, each battery cannot have more (or less) than the maximum (or the minimum) allowed energy, described as Equation 2.9.

$$\forall i \in [1, B^{bat}],\ \forall n \in [1, N^{day}],\ e_i^{min} \leq soc_{i,n} \leq e_i^{max} \tag{2.9}$$

- *SOC constraints*: At the first time interval of the scheduling horizon, each battery must satisfy an initial energy level, defined as Equation 2.10 and 2.11.

$$\forall i \in [1, B^{bat}],\ (soc_{i,1} - e_i^{init}) \times N^{hour} = b_{i,1}^{+} \times \eta_b + b_{i,1}^{-} \tag{2.10}$$

  At each time interval n, the SOC depends on the SOC at time interval $n-1$, and the charge and discharge at time interval n:

$$\forall i \in [1, B^{bat}],\ \forall n \in [2, N^{day}],\ (soc_{i,n} - soc_{i,n-1}) \times N^{hour} = b_{i,n}^{+} \times \eta_b + b_{i,n}^{-} \tag{2.11}$$

Note that, there are various ways to apply round-trip efficiencies when calculating SOCs. For example, some works divided actual charges by the efficiency and multiplied discharges with the efficiency [4, 10]. Some works multiplied both charges and discharges with the efficiency [11], instead. Some other works multiplied charges with the efficiency and divided discharges by the efficiency [12, 13]. This work chooses to multiple charges with the efficiency only as shown in Equation 2.10 and 2.11. However, the use of efficiencies can be changed according to actual needs for different problem instances.

2.4.5 Problem Objectives

The objectives of our battery scheduling problem for peak demand management include minimisation of the peak demand charge, a penalty cost of charging or discharging batteries frequently and a penalty cost of not fully charging the battery before the end of the day. Each objective is calculated as follows:

- Peak demand charge f^{cost}:

$$f^{cost} = \bar{d}^{ann} \times p^{ann} + \bar{d}^{summ} \times p^{summ} \times \alpha \tag{2.12}$$

$$\alpha = \begin{cases} 1, & \text{if the current month is Nov/Dec/Jan/Feb/March,} \\ 0, & \text{otherwise} \end{cases} \tag{2.13}$$

- Penalty cost of frequent charging or discharging $g^{penalty}$:

$$g^{penalty} = \sum_{i=1}^{B^{bat}} \sum_{n=1}^{N^{day}} b_{i,n}^{+} \tag{2.14}$$

- Penalty cost of not fully charging batteries before the end of the day g^{eod}:

$$g^{eod} = \sum_{i=1}^{B^{bat}} (e_i^{max} - soc_{i,n^{eod}}) \tag{2.15}$$

  where n^{eod} is the last time interval of the day.

- Combined objective $h^{combined}$:

$$h^{combined} = w^{cost} \times f^{cost} + w^{penalty} \times g^{penalty} \tag{2.16}$$

  Each objective is multiplied by a weight, denoted as follows:

- The weight of the peak demand charge: w^{cost}.
- The weights of the penalty costs: $w^{penalty}$ and w^{eod}

The values of w^{cost}, $w^{penalty}$ and w^{eod} will affect the contributions of the peak demand charge and the penalty in the combined objective. This work considers w^{cost} to be one. The values of $w^{penalty}$ and w^{eod} were chosen based on observations from experiments.

2.4.6 Formal Problem Formulation

The battery scheduling problem for peak demand management can be formulated as follows:

$$\begin{array}{ll} \textsf{minimise} & h^{combined} \\ \textsf{subject to} & (2.6),\ (2.7),\ (2.8),\ (2.9),\ (2.10),\ (2.11) \end{array}$$

2.5 Battery Scheduling Method for Peak Demand Management

The solution to solving our battery scheduling problem includes a linear programming (LP) model implemented in a modelling language called MiniZinc [14]. In order to incorporate uncertainty in future demand, a rolling horizon rescheduling strategy (RHRS) is applied to update battery schedules regularly during the day based on new demand forecasts. This section describes the linear programming (LP) in Section 2.5.1 and the rolling horizon rescheduling strategy (RHRS) in Section 2.5.2.

2.5.1 Linear Programming Model for Peak Demand Management

Figure 2.5.1 shows the LP model for solving the battery scheduling problem. We declare the input parameters from Line 1 to 24, including:

- The number of time intervals per day: `num_intervals`, line 2,
- The number of time intervals per hour: `num_intervals_hour`, line 3,
- The index of the last time interval of the day: `eod_interval`, line 4,
- The number of batteries: `num_batteries`, line 8,
- The initial SOCs or energy levels: `init_socs`, line 9,
- The minimum SOCs or capacities: `min_socs`, line 10,
- The maximum SOCs or capacities: `max_socs`, line 11,
- The maximum power rates: `max_powers`, line 12,
- The battery efficiencies: `efficiencies`, line 14,
- The demand forecasts: `forecasts`, line 16,
- The peak demand tariff rates: `tariff_rates`, line 19,
- The applicable hours or times of tariffs: `tariff_times`, line 20,
- The pre-defined maximum demand thresholds for tariffs (given by users or calculated from historic demands): `previous_max_demands`, line 22,

Figure 2.1: LP model for solving our battery scheduling problem for peak demand management

```
% ---------- Input parameters ---------- %
int: num_intervals;
int: num_intervals_hour;
int: eod_interval;
set of int: INTERVALS = 1..num_intervals;

int: num_batteries;
set of int: BATTERIES = 1..num_batteries;
array[BATTERIES] of float: init_socs;
array[BATTERIES] of float: min_socs;
array[BATTERIES] of float: max_socs;
array[BATTERIES] of float: max_powers;
float: power_limit = max(max_powers);
array[BATTERIES] of float: efficiencies;

array[INTERVALS] of float: forecasts;
float: demand_limit;

array[int] of float: tariff_rates;
array[int, INTERVALS] of int: tariff_times;
set of int: RATES = index_set(tariff_rates);
array[int] of float: previous_max_demands;
float: w_penalty;
float: w_eod;

% ---------- Variables ---------- %
array[RATES] of var 0..demand_limit: max_demands;
array[BATTERIES, INTERVALS] of var 0..power_limit: charges;
array[BATTERIES, INTERVALS] of var -power_limit..0: discharges;
array[BATTERIES, INTERVALS] of var float: soc;
array[INTERVALS] of var 0..demand_limit: modified_demand = array1d([forecasts[i] + sum(b in BATTERIES)(charges[b, i] + discharges[b, i])| i in INTERVALS]);

% ---------- Constraints ---------- %
% Charge or discharge only constraint
constraint forall(b in BATTERIES, i in INTERVALS) (charges[b, i] * discharges[b, i] = 0);

% Maximum power constraints
constraint forall(b in BATTERIES, i in INTERVALS)(discharges[b, i] <= 0.0);
constraint forall(b in BATTERIES, i in INTERVALS)(discharges[b, i] >= -max_powers[b]);

constraint forall(b in BATTERIES, i in INTERVALS)(charges[b, i] >= 0.0);
constraint forall(b in BATTERIES, i in INTERVALS)(charges[b, i] <= max_powers[b]);

% Maximum-minmum capacity constraints
constraint forall(b in BATTERIES, i in INTERVALS)(soc[b, i] <= max_socs[b]);
constraint forall(b in BATTERIES, i in INTERVALS)(soc[b, i] >= min_socs[b]);

% SOC constraints
constraint forall(b in BATTERIES)
(soc[b, 1] * num_intervals_hour - init_socs[b] * num_intervals_hour =
charges[b, 1] * efficiencies[b] + discharges[b, 1]);

constraint forall(b in BATTERIES, i in 2..num_intervals)
(soc[b, i] * num_intervals_hour - soc[b, i - 1] * num_intervals_hour =
charges[b, i] * efficiencies[b] + discharges[b, i]);

% Max demand calculation
constraint forall(r in RATES)(max_demands[r] >= previous_max_demands[r]);
constraint forall(r in RATES, i in INTERVALS)
(max_demands[r] >= modified_demand[i] * tariff_times[r, i]);

% ---------- Objectives ---------- %
var float: eod_penalty = sum(i in INTERVALS, r in RATES, b in BATTERIES) ((max_socs[b] - soc[b, i]) * (1 - charge_times[r, i]));
var float: penalty = sum(b in BATTERIES, i in INTERVALS)(charges[b, i]);
var float: costs = sum(r in RATES) (max_demands[r] * tariff_rates[r]);
var float: combined_obj = costs + penalty * w_penalty + eod_penalty * w_eod;

solve minimize combined_obj;
```

- The weight for the penalty cost of charging/discharging batteries frequently: `w_penalty`, line 23.
- The weight of the penalty cost of not fully charging batteries before the end of the day: `w_eod`, line 24.

We introduce the variables from line 26 to line 31, including:

- The maximum demand for each tariff: `max_demands`, line 27.
- The amount of power charged to each battery per time interval: `charges`, line 28.
- The amount of power discharged from each battery per time interval: `discharges`, line 29.
- The SOC of each battery per time interval: `socs`, line 30.
- The modified demand per time interval: `modified_demand`, line 31.

We define the constraints from line 33 to line 60, including:

- The *charge or discharge only constraint*: line 34.
- The *maximum power constraint*: line 37.
- The *maximum-minimum capacity constraint*: line 44.
- The *SOC constraints*: line 48.
- The *maximum demand constraints* for calculating the maximum demand for each peak demand tariff: line 57.

We specify the objective functions from line 62 to line 66, including:

- The penalty cost of not fully charging batteries before the end of the day: `eod_penalty`, line 63.
- The penalty cost of charging/discharging batteries frequently: `penalty`, line 64.
- The total peak demand charges: `costs`, line 65.
- The combined objective: `combined_obj`, line 66.

2.5.2 Rolling Horizon Rescheduling Strategy

The RHRS is a method that incorporates uncertainty in demand forecasts. This strategy applies the optimal solution to an optimisation problem for a period of time, after which the values of the input parameters are updated and new optimal solution is obtained [6]. More specifically, this strategy solves a problem in an iterative manner as follows:

1. At any time interval of the day n, the battery scheduling problem is solved for the entire scheduling horizon, which can be from the current time interval until the last time interval of the day, or the next 24 hours.

2. Use the optimal decisions for time intervals n, $n+1$, ..., $n+x$ to control the batteries.

3. At time interval $n+x$, the battery scheduling problem is solved again for the entire scheduling horizon, and Step 1 and Step 2 are repeated every x intervals until the last time interval of the day.

This iterative process can be illustrated as Figure 2.2. At each iteration, the maximum demand for each tariff calculated in the previous iteration will be used as the pre-defined maximum demand thresholds for the current iteration. If the forecast maximum demand is lower than the pre-defined maximum demand thresholds at any time interval or iteration, the battery scheduling problem does not need to be solved again for that iteration.

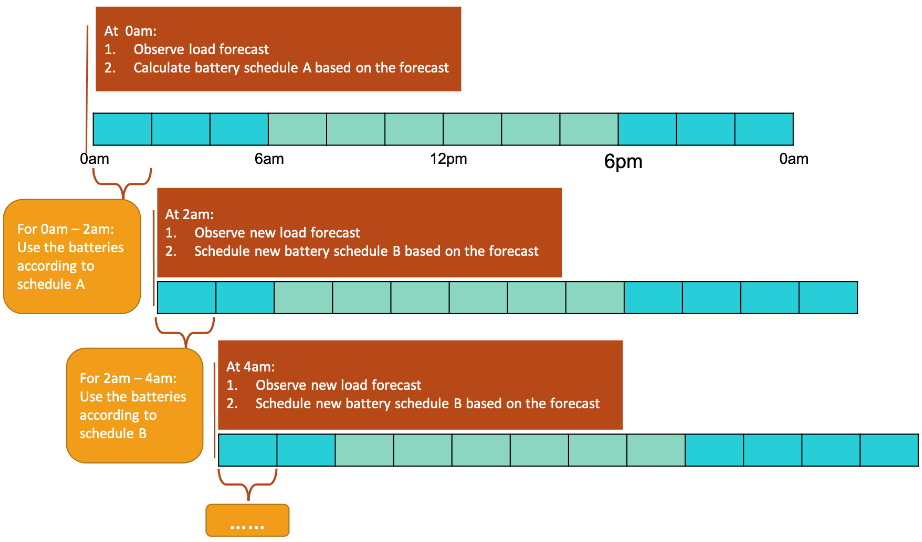

Figure 2.2: Rolling horizon rescheduling strategy

2.6 Experimental Results of Battery Scheduling Method for Peak Demand Management

This section presents the experiment environments, steps and results that demonstrate the effectiveness of our battery scheduling method for peak demand management.

2.6.1 Experiment Environment

The linear programming (LP) model was implemented in MiniZinc [14], which is an optimisation language accepted by well-known optimisation solvers, including Gurobi and CPLEX. The rolling horizon rescheduling strategy (RHRS) was programmed in Python. Details of the implementation can be found at `https://bitbucket.org/dorahee2/battery-scheduling/src/master/`. The instruction of using this algorithm has been explained a wiki at `https://bitbucket.org/dorahee2/battery-scheduling/wiki/Home`. At the current stage of the project, this repository needs to remain private. Please email `dora.he3@monash.edu` for access to this repository and the wiki.

2.6.2 Experiment Data

The data for testing the battery scheduling method for peak demand management includes:

- Peak demand tariffs, which have described in Section 2.2.
- Battery details: Two batteries are used in this work.
  - A Lithium-ion battery (Li-ionB) whose maximum power rate is 120 kW, the maximum capacity is 134 kWh, and the round-trip efficiency is between 85% and 88%. We have chosen 88% as the efficiency for our experiments.
  - A Vanadium Flow battery (VFB) whose maximum power rate is 180 kW, the maximum capacity is 900 kWh and the round-trip efficiency is between 60% and 65%. We have chosen 65% as the efficiency for our experiments.
- Historic demands are averaged demands at 30-minute intervals from 1st January 2020 00:00 to 31st December 2020 23:00.
- Demand forecasts are produced at every 30-minute interval from 1st January 2020 00:00 to 31st December 2020 23:00. Each forecast has the predicted average load for every 30-minute interval in the next 24 hours.

At the time of this work, the forecast results from Farshid's model were not available for testing. Frits has developed a naive machine learning model to forecast demands for these experiments. In practise, demand forecasts from any machine learning model can be used.

The details of the input data including the expected values and formats of these data are explained in `https://bitbucket.org/dorahee2/battery-scheduling/wiki/Peak%20demand%20management/2.%20Input%20data`.

2.6.3 Experiment Results

We have tested the effectiveness of our battery scheduling method for peak demand management using the data from 1 Jan 2020 to 30 Dec 2020 in the following step:

1. Set the time window to be from 6am to 0am on 1 Jan 2020.
2. Read the demand forecast of this time window and schedule batteries for this time window based on the forecast.
3. Select the frequency at which the forecast is updated, such as every 24/12/6/3 hours or 30 minutes.
4. Apply the optimised battery schedule for the duration of the frequency. For example, add the battery charges and discharges to the actual demands for the rest of the day, or for the next 12/6/3 hours or 30 minutes.
5. Move the time window for the frequency length. For example, move the time window to start from the next day, or from 12/6/3 hours or 30 minutes after 6am.
6. Repeat Step 2 – Step 5 at the chosen frequency until the time window reaches the last time interval on 30 Dec 2020.

Note that, in this work, the time window always starts from a time interval in a day and finishes at the last time interval of the same day. The size of the time window reduces gradually during the day and resets in the next day. However, in practice, this time window can maintain a fixed size. For example, the size of the time window can remain 24 hours.

We have scheduled batteries using data from 1 Jan 2020 to 30 Dec 2020 and evaluated results using different forecasts and forecasting frequencies, as follows:

1. Perfect forecast scenario: Using historic demands as the demand forecasts.
2. Forecasts updated every day scenario: Using the naive demand forecast model and updating the forecast once at the beginning of the day.

3. Forecast updated every 12 hours: Using the naive demand forecast model and updating the forecast every 12 hours.

4. Forecast updated every 6 hours: Using the naive demand forecast model and updating the forecast every 6 hours.

5. Forecast updated every 3 hours: Using the naive demand forecast model and updating the forecast every 3 hours.

6. Forecast updated every 30 minutes: Using the naive demand forecast model and updating the forecast every 30 minutes.

We have illustrated the forecast demands, the actual demands, the maximum demand associated with the annual peak demand charge, the maximum demand associated with the summer monthly peak demand charge, the charges and discharges of each battery for January of 2020 in each scenario in Figure 2.3, 2.4, 2.5, 2.6, 2.7.

We have also calculated the optimised annual and monthly peak demand charges for 2020 in each scenario, and compared them with the original charges when no batteries are used in Table 2.6.4 and 2.6.4.

The results show that the batteries discharged at the maximum power rates during peak times and recharged gradually outside those times when the demand forecasts were perfect, or the demand forecasts followed the trend of the actual demands, and therefore minimising or reducing the peak demand charges. However, the batteries may recharge at the undesired times or not discharge at the desired times when the forecasts were not accurate. In the worst case, the peak demand charges may be higher than those without using batteries.

2.6.4 Findings

We have found that this battery scheduling method can effectively reduce the peak demand charges when demand forecasts indeed approximate actual demands. However, inaccurate demand forecasts can lead to a higher peak demand charge in the worst case.

Table 2.1: The actual and optimised peak demand charges

	Annual	**Jan**	**Feb**	**Mar**	**Nov**	**Dec**
No optimisation	$ 2,212.56	$ 2,686.32	$ 2,230.80	$ 2,064.40	$ 1,758.64	$ 1,810.64
Optimisation with forecasts						
Perfect forecasts	$ 2,173.05	$ 2,326.61	$ 2,182.05	$ 2,015.65	$ 1,709.89	$ 1,761.89
Updating forecasts every day	$ 2,212.56	$ 2,375.36	$ 2,230.80	$ 2,064.40	$ 1,718.39	$ 1,804.35
Updating forecasts every 12 hours	$ 2,212.56	$ 2,375.36	$ 2,230.80	$ 2,064.40	$ 1,718.39	$ 1,804.35
Updating forecasts every 6 hours	$ 2,212.56	$ 2,375.36	$ 2,206.95	$ 2,017.12	$ 1,729.48	$ 1,787.37
Updating forecasts every 3 hours	$ 2,212.56	$ 2,637.57	$ 2,182.05	$ 2,017.12	$ 1,709.89	$ 1,761.89
Updating forecasts every hour	$ 2,236.27	$ 2,637.57	$ 2,182.05	$ 2,015.65	$ 1,709.89	$ 1,761.89
Updating forecasts every 30 minutes	$ 2,240.23	$ 2,637.57	$ 2,182.05	$ 2,015.65	$ 1,709.89	$ 1,761.89

Table 2.2: The reductions of the optimised peak demand charges

Forecast type	**Annual**	**Jan**	**Feb**	**Mar**	**Nov**	**Dec**
Perfect forecasts	**2%**	**13%**	2%	2%	3%	3%
Updating forecasts every day	0%	12%	0%	0%	2%	0%
Updating forecasts every 12 hours	0%	12%	0%	0%	2%	0%
Updating forecasts every 6 hours	0%	12%	1%	2%	2%	1%
Updating forecasts every 3 hours	0%	2%	2%	2%	3%	3%
Updating forecasts every hour	-1%	2%	2%	2%	3%	3%
Updating forecasts every 30 minutes	-1%	2%	2%	2%	3%	3%

2.7 Conclusion

This work has developed a method that schedules batteries based on demand forecasts to reduce the network peak demand charge (NPDC) for Monash Clayton campus. This method combines linear programming (LP) and rolling horizon rescheduling strategy (RHRS) to decide the best amount of power to charge and discharge for each battery at each time while incorporating uncertainty in the future demand. The experiment results have found that this method is effective at reducing the peak demand charges when the demand forecasts indeed approximate the actual demands.

In order to hedge against the risks of uncertain future and inaccurate forecasts, future work can consider employing additional methods to incorporate various possible future scenarios, and scheduling batteries based on the expected outcomes of those future scenarios instead of a single forecast. Other future work can include investigating the scheduling results using different sizes of batteries, and evaluating the economic costs and returns of those battery sizes.

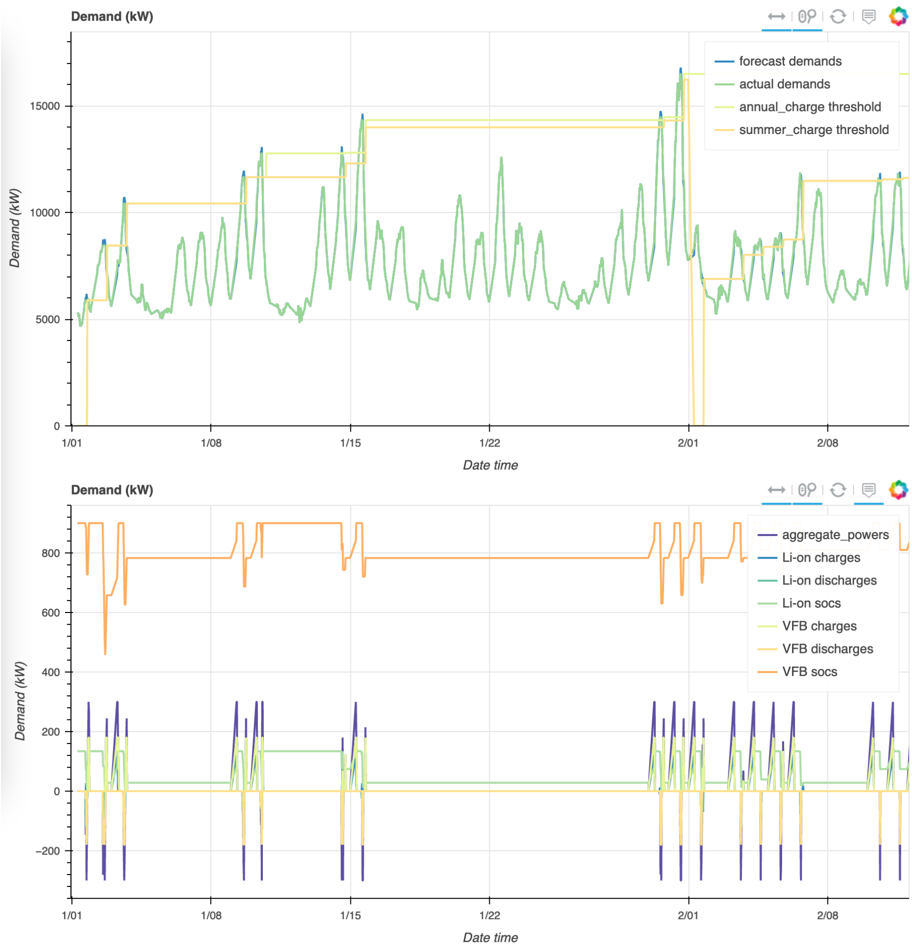

Figure 2.3: Optimisation results with perfect forecasts

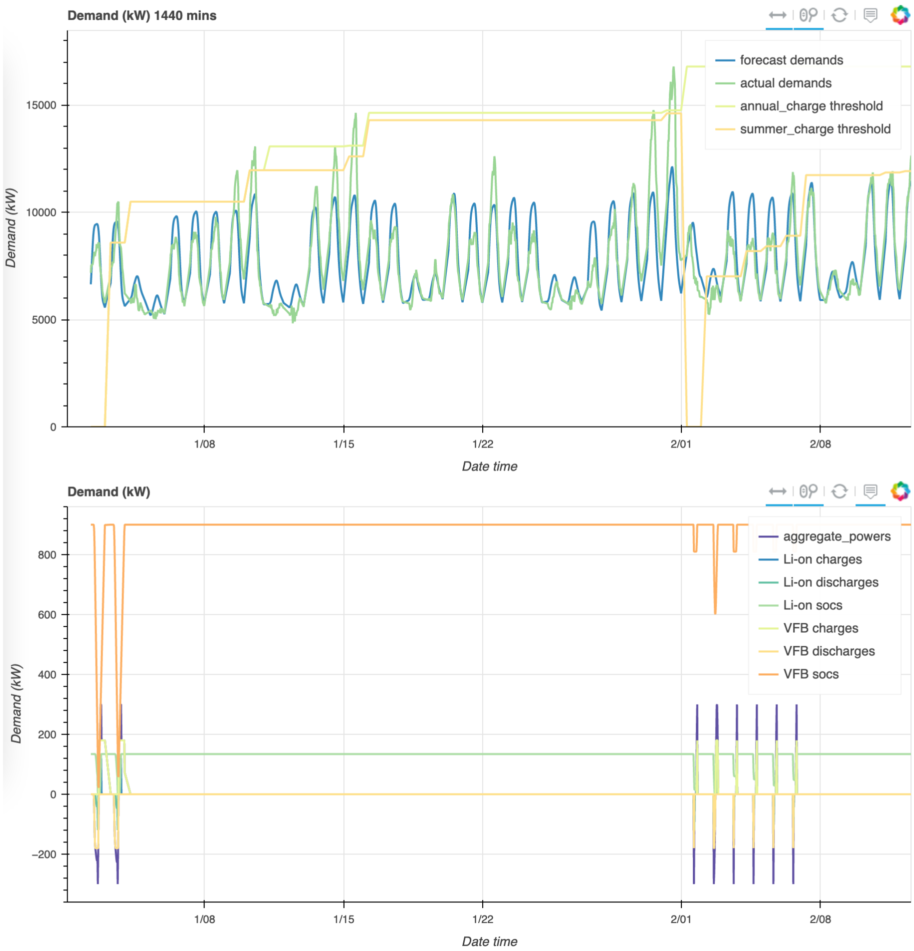

Figure 2.4: Optimisation results with forecasts that are updated every day

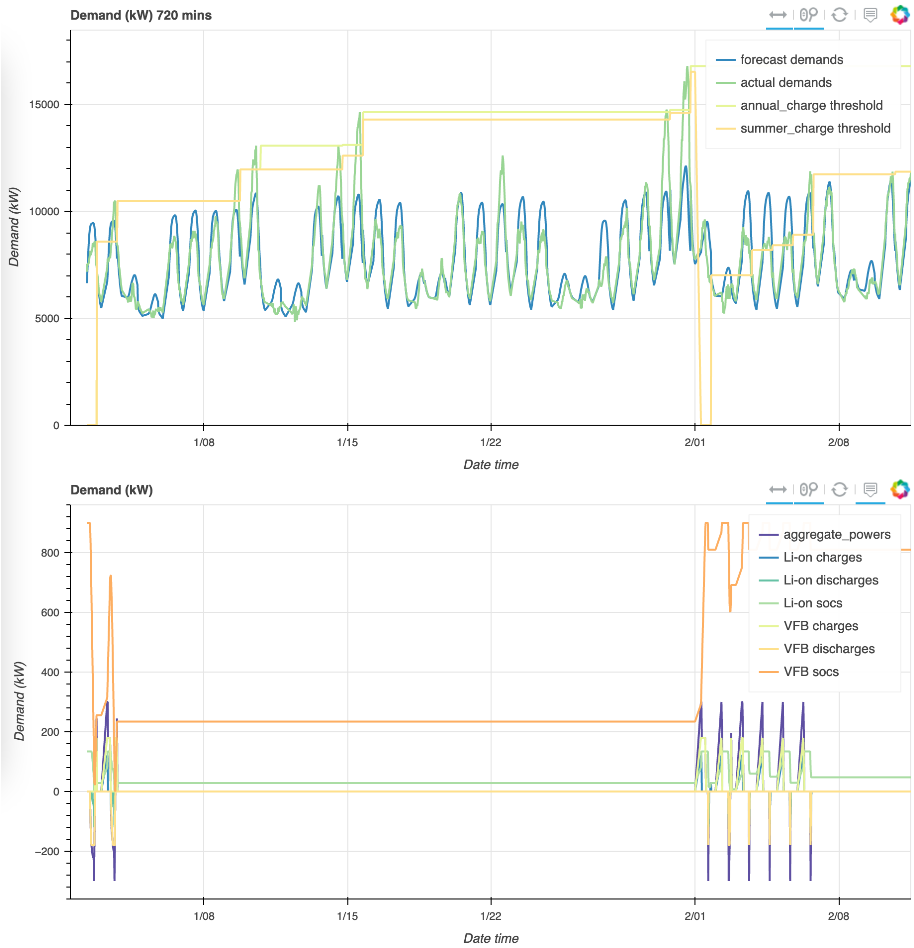

Figure 2.5: Optimisation results with forecasts that are updated every 12 hours

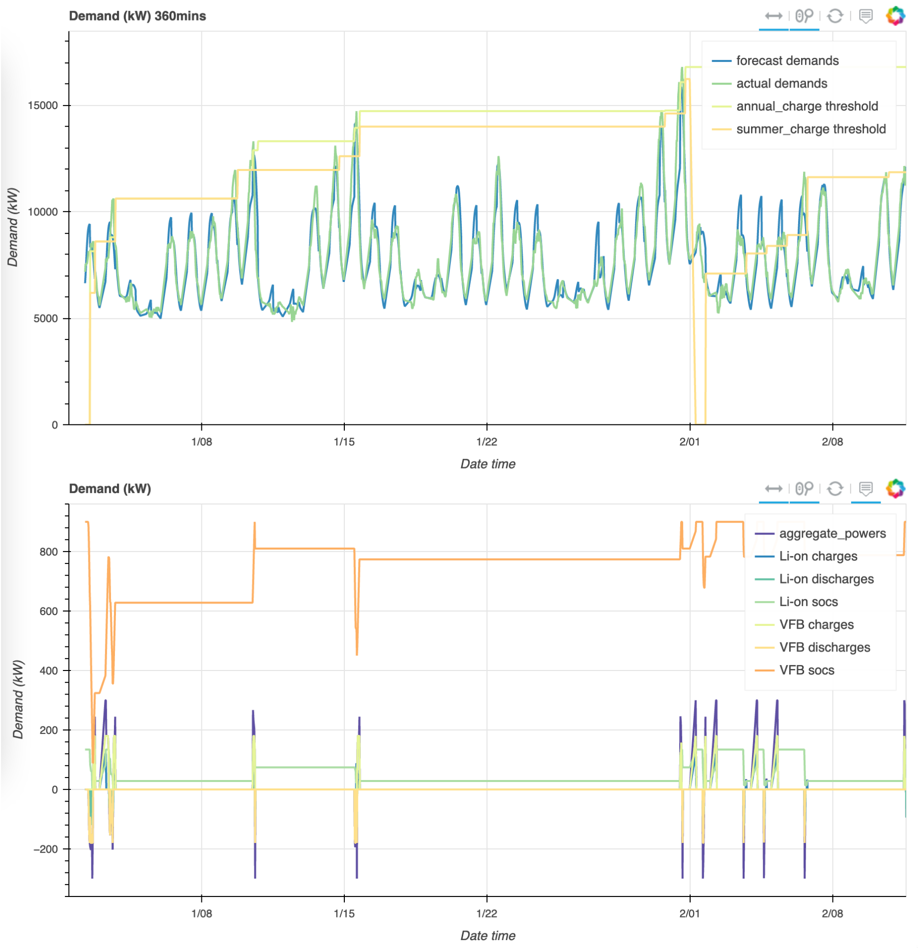

Figure 2.6: Optimisation results with forecasts that are updated every 6 hours

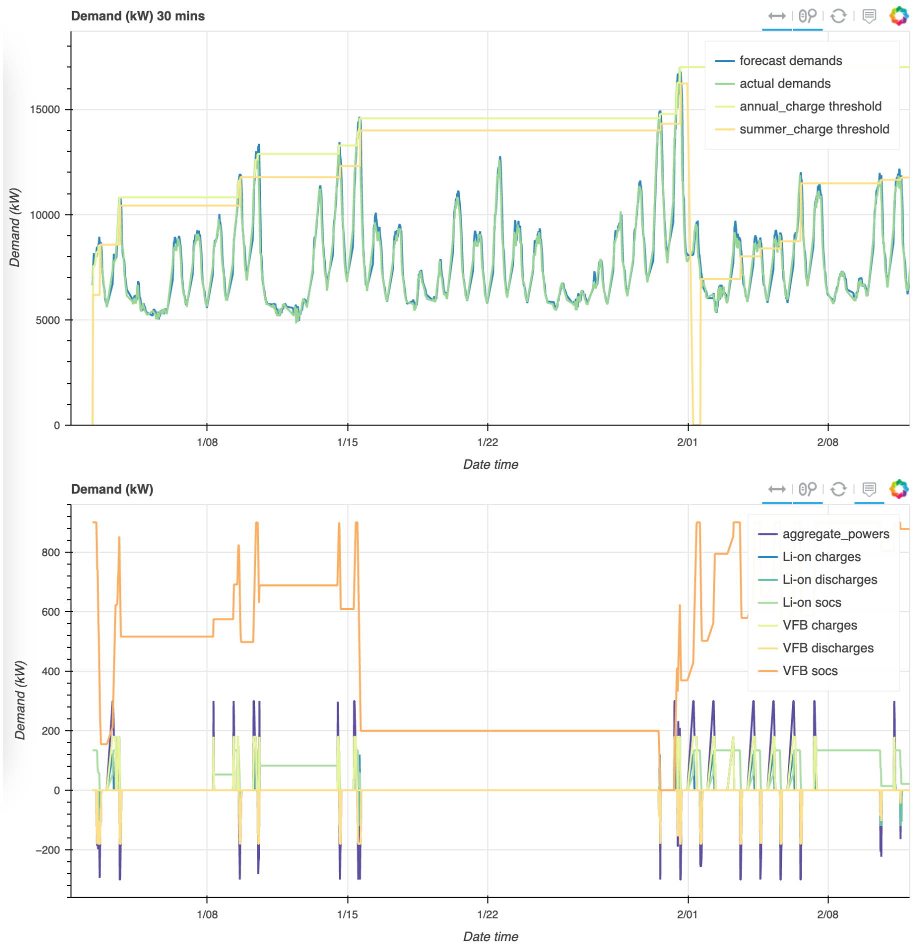

Figure 2.7: Optimisation results with forecasts that are updated every 30 minutes

Chapter 3

Demand Response

3.1 Summary

Large consumers can participate in *demand response* to receive financial rewards by reducing consumption at times with high electricity wholesale prices. One way to earn rewards without changing existing consumption patterns is to use batteries. We can charge batteries during times with low prices and discharge them at times with high prices. This work develops a scheduling algorithm that decides the best times to charge and discharge batteries, in order to maximise the rewards. This algorithm combines linear programming and rolling horizon rescheduling strategy to use demand and price forecasts for scheduling while considering uncertainty in these forecasts. Further works are still required to evaluate the results of this algorithm.

3.2 Demand Response Financial Rewards

Demand response (DR) refers to activities of reducing loads in response to financial incentives. Monash receives financial rewards for reducing loads when the wholesale price is above a threshold via contracts with the retailer. Monash will retain 60% of the wholesale price above this trigger price for load reduction delivered beyond the adjusted baseline load, with the retailer retaining the other 40%. Several conditions are required for Monash to meet, in order to receive financial rewards from DR, including:

- Meeting trigger price: Load reductions should occur when the wholesale price is above the trigger price, which is currently defined at $300 / MWh for the Clayton campus.

- Meeting the minimum load reduction: Monash should reduce at least 1 kWh of the loads in order to receive rewards.
- Defining the load measurements: The loads should be measured at 15-minute intervals, however, the load reductions are calculated based on the 30-minute averages.

3.2.1 Calculation of Financial Rewards

The financial reward is calculated for each trading interval where the DR conditions are met. The duration of a trading interval was thirty (30) minutes before Oct 2021, however, is now five (5) minutes. In respect to a trading interval n, the load reduction financial benefit is based on the wholesale spot price, the adjusted baseline load, the measured or the actual load, a *load reduction value (LRV)* and a loss factor, described as Equation 3.1.

$$\forall n,\ r_n = p_n^{spot} \times min(0, (l_n^{bl} - l_n^{actual})) \times LRV \times Lossfactor \tag{3.1}$$

where:

- r_n is the reward for the trading interval n,
- p_n^{spot} is the wholesale spot price for the trading interval n,
- l_n^{bl} is the baseline load for the trading interval n in MWh,
- l_n^{actual} is the measured or the actual load for the trading interval n in MWh,

The wholesale spot prices are provided by Australia Electricity Market Operator (AEMO). The actual loads are measured by electricity meters. The LRV is set to be 60% for Monash. The loss factor is a fixed value (about 0.9) that applies to the specified NMI (ID: VEEE08KH3V), which is neglect-able in this work. The adjusted baseline load needs to be calculated from historical loads.

3.2.2 Calculation of Baseline Loads

The baseline load with the respect to a specified NMI and a trading interval is calculated using:

- The average amount, in MWh, of the measured load (with respect to the specified NMI) for the ten (10) trading intervals occurring at the same time of day as that trading interval on the ten (10) like days immediately preceding on which that trading interval occurs;

- The relevant uplift factor for the corresponding maximum temperature of the day.

The baseline load for a trading interval of a day is calculated from the loads of the same trading interval in the prior ten like days and a uplift factor that corresponds to the maximum temperature of the day.

Like days

A like day is defined as follows:

- With respect to a trading interval occurring on a Monash work day, a like day should be a Monash work day.
- With respect to a trading interval occurring on a non Monash work day, a like day should be a non Monash work day. For example, if an interval on a Saturday has a spot price greater than the trigger price $300 / MWh, then the baseline is set using the former five (5) Saturdays and five (5) Sundays (ten days total) assuming no public holidays occurred in that period.
- Like days exclude any day on which **any** trading interval has occurred with respect to which a payment needs to be made. For example, if a DR load reduction event has occurred in any interval on one of the prior ten (10) like-days that would ordinarily be used to calculate a baseline, and Monash successfully actioned a load reduction such that the retailer was subject to making a payment for that interval on that day, then **that whole day** is excluded from the baseline, and the next prior like-day needs to be used to set the respective baseline.

The definition of Monash work days is provided in `https://www.monash.edu/students/admin/dates/holidays`.

Uplift Factor

The uplift factors corresponding to maximum temperatures are listed as Table 3.1.

3.3 Methodology of Battery Scheduling for Demand Response

We can use battery energy storage systems (batteries) to discharge during intervals with wholesale prices that are higher than the trigger price, and charge outside those periods, in order to earn financial rewards without changing the existing consumption patterns.

Similar to the method developed in Chapter 2, the work in this chapter follows the methodology explained in Section 1.2. More specifically, this work combines *linear programming (LP)* models and *rolling horizon rescheduling strategy (RHRS)* to optimally schedule batteries and incorporate uncertainty in demand and price forecasts.

3.4 Battery Scheduling Problem for Demand Response

This work studies a battery scheduling problem where batteries are scheduled to maximise the financial rewards from DR. This section presents the input parameters, variables, constraints, objective functions and mathematical formulation of the battery scheduling problem.

3.4.1 Problem Parameters

The input parameters for this battery scheduling problem include time intervals, battery specifications, parameters related to demand response rewards, load forecasts, historic loads, price forecasts, historic prices and objective weights.

Time Interval

A day is considered as a finite number of time intervals for scheduling. We assume that the charge or discharge rate of a battery does not change in a time interval. Let us write:

- T as the length in minutes of each time interval,
- $N^{day} = 1440/T$ as the total number of time intervals per day,
- $N^{hour} = 60/T$ as the total number of time intervals per hour,

We have chosen $T = 30$ mins for this work, because at the time of this work, the wholesale electricity prices are updated every thirty minutes.

Battery Specification

The total number of batteries is denoted as B^{bat}. Each battery $i \in [1, B^{bat}]$ is modelled by the following elements:

- e_i^{init}: the initial energy level at the beginning of the day (in kWh)

- e_i^{min}: the minimum allowed energy capacity (in kWh)
- e_i^{max}: the maximum allowed energy capacity (in kWh)
- $\bar{p}_i$: the maximum power rate (in kW)
- $b_{i,n}^{+}$: the amount of power charged to the battery at time step n (in kW):
- $b_{i,n}^{-}$: the amount of power discharged from the battery at time step n (in kW)
- η_b: the round-trip efficiency (between 0 and 1)
- $soc_{i,n}$: a state-of-charge (SOC) profile — the amount of energy remaining in the battery at time step n (in kWh)

Demand Response Reward Parameters

The parameters required for calculating the demand response rewards include:

- Trigger price $p^{trigger}$: An essential condition for receiving the demand response rewards, as explained in Section 3.2.
- Forecast maximum temperature: The maximum temperature forecasted at the time of the demand forecast.
- Uplift factors α^{uplift}: The uplift factor associated with the maximum forecasted temperature for scaling the baseline loads, as explained in Section 3.2.2.
- Load reduction value LRV: A value set for Monash, as explained in Section 3.2.1.
- Loss factor: A factor specific to the electricity meter used for measuring the actual loads, as explained in Section 3.2.1.

In addition, we have introduced binary variables that indicate the time intervals whose forecast spot prices are above the trigger price. Theses binary variables are defined as $\mathbf{t}^{dr} = \{t_n^{dr} = 1 \; if \; p_n^{spot} \geq p^{trigger} \; else \; 0 \mid n \in [1, N^{day}]\}$.

Load Forecast

Load forecasts are required to plan the battery operations in advance. At the time of this work, at each time interval n, the demand is forecasted from the upcoming time interval $n+1$ until 6pm of the same day. However, to allow batteries to charge after 6pm, we assume the forecasted demand after 6pm is zero. Let us write l_n^{fore} as the load per time interval in kWh.

Historic Load

Historic loads are used for calculating the baseline loads from like-days. Ideally, at each time interval n, the historic loads should be for the three months prior to the current time interval. If less than three months of data is provided, the baseline loads will be calculated from the available data only. Let us write the historic loads at time interval n of day d as $l_{d,n}^{his}$.

Price Forecast

Price forecasts are required to prepare batteries for potential demand response opportunities. The price forecasts are provided by AEMO. Let us write p_n^{fore} as the forecasted spot price per time interval in $/MWh.

Historic Price

Historic prices are used for identifying past demand response event days, and excluding those days from like days for calculating baseline demands. Let us write $p_{d,n}^{his}$ as the historic spot price per time interval in $/MWh of day d.

Objective Weight

This work has considered multiple objectives including maximisation of the financial reward and minimisation of a penalty cost of charging or discharging batteries frequently. The penalty cost is introduced to prevent batteries from fluctuating between charging and discharging during the day. The objective weights are denoted as:

- The weight of the financial reward: w^{reward}.
- The weight of the penalty cost: $w^{penalty}$.

3.4.2 Problem Variables

The decision variables include the amount of power charged to or discharged from each battery at each time interval: $b_{i,n}^{+}$ and $b_{i,n}^{-}$.

3.4.3 Problem Constraints

The constraints are the same as those in Section 2.4.4 in Chapter 2.

3.4.4 Problem Objectives

The objectives of our battery scheduling problem for demand response include maximisation of the financial reward and minimisation of a penalty cost of charging or discharging batteries frequently. The penalty cost $g^{penalty}$ is the same as that in Section 2.4.5. The financial reward objective is calculated as Equation 3.1. The combined objective $h^{combined}$ for demand response is described as

$$h^{combined} = w^{reward} \times \sum_{n=1}^{N^{day}} r_n \times t_n^{dr} + w^{penalty} \times g^{penalty} \tag{3.2}$$

3.4.5 Formal Problem Formulation

The battery scheduling problem for demand response can be formulated as follows:

$$\begin{aligned} \text{minimise} \quad & h^{combined} \\ \text{subject to} \quad & (2.6),\ (2.7),\ (2.8),\ (2.9),\ (2.10),\ (2.11) \end{aligned}$$

3.5 Battery Scheduling Method for Demand Response

Same as the solution presented in Section 2.5 of Chapter 2, the solution to battery scheduling for demand response uses a linear programming (LP) model and a rolling horizon rescheduling strategy (RHRS). The details of rolling horizon rescheduling strategy (RHRS) has been presented in Section 2.5. This section focuses on the algorithm for calculating the baseline loads in Section 3.5.1 and the linear programming (LP) model for maximising demand response rewards in Section 3.5.2.

3.5.1 Baseline Load Calculation

Currently the baseline load is calculated using historic loads only without historic prices, due to the absence of sufficient historic prices for testing. This means, past DR event days are not excluded.

Algorithm 1 Calculation of Baseline Loads

Require: Historic loads
1: **for all** time interval n in the scheduliung horizon **do**
2: Identify like days using the conditions introduced in Section 3.2.2.
3: Sum up the loads at the same interval of the past ten like days.
4: Calculate the average load of the ten like days at time interval n.
Ensure: The average load per time interval in the scheduling horizon.

3.5.2 Linear Programming Model for Demand Response

Figure 3.1 shows the LP model for solving the battery scheduling problem for demand response. We declare the input parameters from Line 1 to 25, including:

- The number of time intervals per day: `num_intervals`, line 2,
- The number of time intervals per hour: `num_intervals_hour`, line 3,
- The index of the last time interval of the day: `eod_interval`, line 4,
- The number of batteries: `num_batteries`, line 8,
- The initial SOCs or energy levels: `init_socs`, line 9,
- The minimum SOCs or capacities: `min_socs`, line 10,
- The maximum SOCs or capacities: `max_socs`, line 11,
- The maximum power rates: `max_powers`, line 12,
- The battery efficiencies: `efficiencies`, line 14,
- The load forecast: `forecast_loads`, line 16,
- The baseline loads: `baseline_loads`, line 18,
- The price forecast: `forecast_prices`, line 19,
- The load reduction value: `load_reduction_value`, line 22,
- The loss factor: `loss_factor`, line 23,
- The weight for the penalty cost of charging/discharging batteries frequently: `w_penalty`, line 24.
- The weight of the penalty cost of not fully charging batteries before the end of the day: `w_eod`, line 25.

We introduce the variables from line 27 to line 31, including:

- The amount of power charged to each battery per time interval: `charges`, line 28.
- The amount of power discharged from each battery per time interval: `discharges`, line 29.
- The SOC of each battery per time interval: `socs`, line 30.
- The modified load per time interval: `modified_loads`, line 31.

Figure 3.1: LP model for solving our battery scheduling problem for demand response

```
% ---------- Input parameters ---------- %
int: num_intervals;
int: num_intervals_hour;
int: eod_interval;
set of int: INTERVALS = 1..num_intervals;

int: num_batteries;
set of int: BATTERIES = 1..num_batteries;
array[BATTERIES] of float: init_socs;
array[BATTERIES] of float: min_socs;
array[BATTERIES] of float: max_socs;
array[BATTERIES] of float: max_powers;
float: power_limit = max(max_powers);
array[BATTERIES] of float: efficiencies;

array[INTERVALS] of float: forecast_loads;
float: load_limit;
array[INTERVALS] of float: baseline_loads;
array[INTERVALS] of float: forecast_prices;

% dr charges
float: load_reduction_value;
float: loss_factor;
float: w_penalty;
float: w_eod;

% ---------- Variables ---------- %
array[BATTERIES, INTERVALS] of var 0..power_limit: charges;
array[BATTERIES, INTERVALS] of var -power_limit..0: discharges;
array[BATTERIES, INTERVALS] of var float: soc;
array[INTERVALS] of var 0..demand_limit: modified_loads = array1d([forecast_loads[i] + sum(b in BATTERIES)(charges[b, i] + discharges[b, i])| i in INTERVALS]);

% ---------- Constraints ---------- %
% Charge or discharge only constraint
constraint forall(b in BATTERIES, i in INTERVALS)(charges[b, i] * discharges[b, i] = 0);

% Maximum power constraints
constraint forall(b in BATTERIES, i in INTERVALS)(discharges[b, i] <= 0.0);
constraint forall(b in BATTERIES, i in INTERVALS)(discharges[b, i] >= -max_powers[b]);

constraint forall(b in BATTERIES, i in INTERVALS)(charges[b, i] >= 0.0);
constraint forall(b in BATTERIES, i in INTERVALS)(charges[b, i] <= max_powers[b]);

% Maximum-minmum capacity constraints
constraint forall(b in BATTERIES, i in INTERVALS)(soc[b, i] <= max_socs[b]);
constraint forall(b in BATTERIES, i in INTERVALS)(soc[b, i] >= min_socs[b]);

% SOC constraints
constraint forall(b in BATTERIES)
(soc[b, 1] * num_intervals_hour - init_socs[b] * num_intervals_hour =
charges[b, 1] * efficiencies[b] + discharges[b, 1]);

constraint forall(b in BATTERIES, i in 2..num_intervals)
(soc[b, i] * num_intervals_hour - soc[b, i - 1] * num_intervals_hour =
charges[b, i] * efficiencies[b] + discharges[b, i]);

% ---------- Objectives ---------- %
var float: eod_penalty = sum(i in INTERVALS, r in RATES, b in BATTERIES) ((max_socs[b] - soc[b, i]) * (1 - charge_times[r, i]));
var float: penalty = sum(b in BATTERIES, i in INTERVALS)(charges[b, i]);
var float: returns = - sum(i in INTERVALS)
(forecast_prices[i] * load_reduction_value * loss_factor *
(baseline_loads[i] - modified_loads[i]));
var float: combined_obj = costs + penalty * w_penalty + eod_penalty * w_eod;

solve minimize combined_obj;
```

We define the constraints from line 33 to line 55, including:

- The *charge or discharge only constraint*: line 34.
- The *maximum power constraint*: line 37.
- The *maximum-minimum capacity constraint*: line 44.
- The *SOC constraints*: line 48.

We specify the objective functions from line 57 to line 63, including:

- The penalty cost of not fully charging batteries before the end of the day: `penalty`, line 59.
- The penalty cost of charging/discharging batteries frequently: `penalty`, line 59.
- The financial rewards: `returns`, line 62.
- The combined objective: `combined_obj`, line 63.

Note that the trigger price needs to be multiplied by 1000 if it is in $ / MWh, because the loads and the battery charges/discharges are in kWh or kW.

3.6 Experimental Results of Battery Scheduling Method for Demand Response

This section presents the experiment environments, steps and results that demonstrate the effectiveness of our battery scheduling method for demand response.

3.6.1 Experiment Environment

The linear programming (LP) model was implemented in MiniZinc [14], which is an optimisation language accepted by well-known optimisation solvers, including Gurobi and CPLEX. The rolling horizon rescheduling strategy (RHRS) was programmed in Python. Details of the implementation can be found at `https://bitbucket.org/dorahee2/battery-scheduling/src/master/`. The instruction of using this algorithm has been explained a wiki at `https://bitbucket.org/dorahee2/battery-scheduling/wiki/Home`. At the current stage of the project, this repository needs to remain private. Please email `dora.he3@monash.edu` for access to this repository and the wiki.

3.6.2 Experiment Data

The data for testing the battery scheduling method for demand response includes:

- Trigger price at \$300 / MWh, which have described in Section 3.2.
- Battery details: Two batteries are used in this work.
  - A Lithium-ion battery (Li-ionB) whose maximum power rate is 120 kW, the maximum capacity is 134 kWh, and the round-trip efficiency is between 85% and 88%. We have chosen 88% as the efficiency for our experiments.
  - A Vanadium Flow battery (VFB) whose maximum power rate is 180 kW, the maximum capacity is 900 kWh and the round-trip efficiency is between 60% and 65%. We have chosen 65% as the efficiency for our experiments.
- Historic loads are averaged loads at 30-minute intervals from 1st January 2020 00:00 to 31st December 2020 23:00.
- Load forecasts are produced at every 30-minute interval from 1st January 2020 00:00 to 31st December 2020 23:00. Each forecast has the predicted average load for every 30-minute interval in the next 24 hours.

At the time of this work, the forecast results from Farshid's model were not available for testing. Frits has developed a naive machine learning model to forecast demands for these experiments. In practise, demand forecasts from any machine learning model can be used.

The details of the input data including the expected values and formats of these data are explained in

3.6.3 Experiment Results

We have tested the effectiveness of our battery scheduling method for demand response using the data from 1 Jan 2020 to 30 Dec 2020 in the following step:

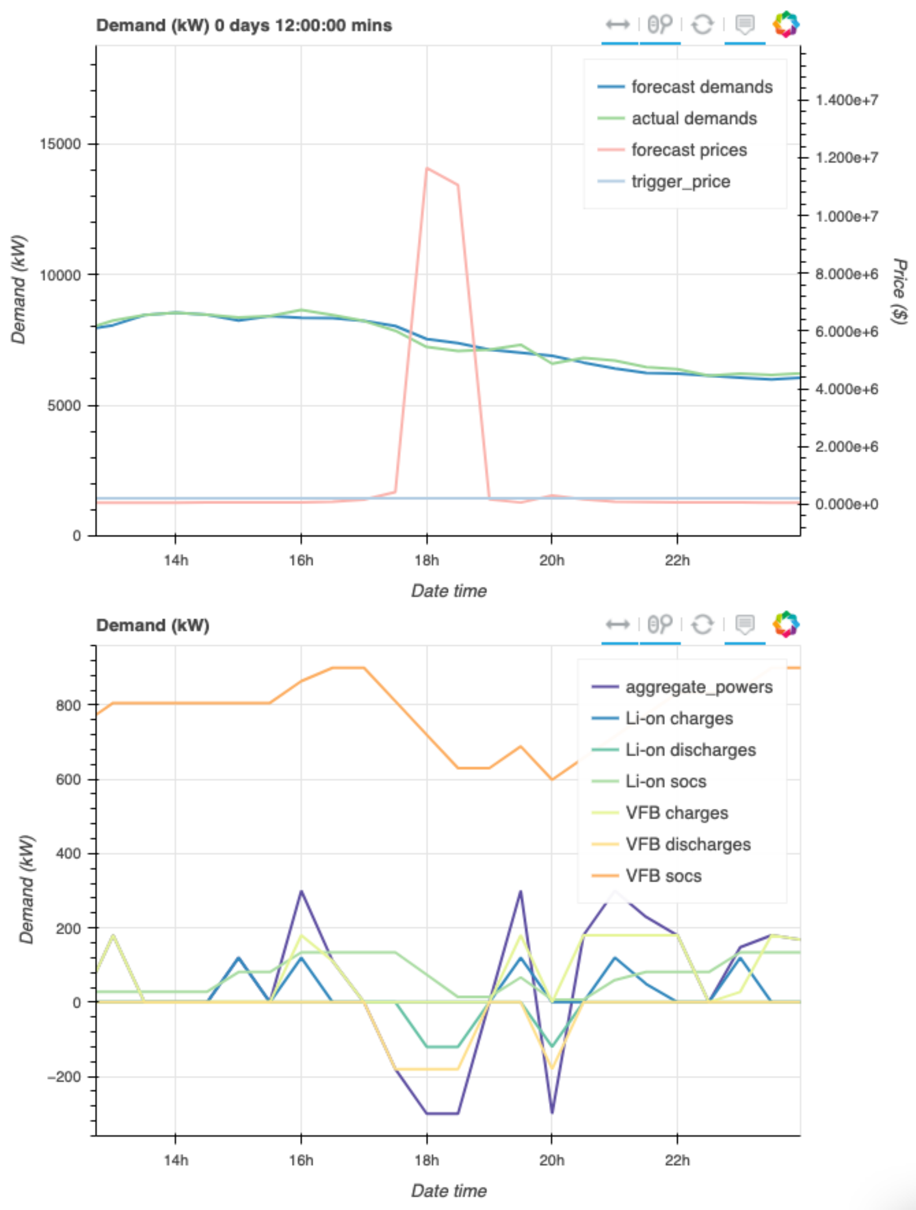
Demand (kW) 0 days 12:00:00 mins
forecast demands
actual demands
forecast prices
trigger_price
Demand (kW)
Price ($)
15000
10000
5000
0
1.400e+7
1.200e+7
1.000e+7
8.000e+6
6.000e+6
4.000e+6
2.000e+6
0.000e+0
14h
16h
18h
20h
22h
Date time
Demand (kW)
aggregate_powers
Li-on charges
Li-on discharges
Li-on socs
VFB charges
VFB discharges
VFB socs
Demand (kW)
800
600
400
200
0
−200
14h
16h
18h
20h
22h
Date time

Figure 3.2:

Table 3.1: Uplift factors for maximum temperature values

Maximum Temperature	**Uplift Factor**
27	106%
28	106%
29	106%
30	116%
31	116%
32	116%
33	116%
34	122%
35	122%
36	122%
37	122%
38	127%
39	127%
40	127%
≥ 40	127%

Bibliography

[1] M. Marzband, H. Alavi, S. S. Ghazimirsaeid, H. Uppal, and T. Fernando, "Optimal energy management system based on stochastic approach for a home microgrid with integrated responsive load demand and energy storage," *Sustainable Cities and Society*, vol. 28, pp. 256 – 264, 2017.

[2] A. Karimi, F. Aminifar, A. Fereidunian, and H. Lesani, "Energy storage allocation in wind integrated distribution networks: An milp-based approach," *Renewable Energy*, vol. 134, pp. 1042 – 1055, 2019.

[3] Y. Li, Z. Yang, G. Li, D. Zhao, and W. Tian, "Optimal scheduling of an isolated microgrid with battery storage considering load and renewable generation uncertainties," *IEEE Transactions on Industrial Electronics*, vol. 66, pp. 1565–1575, Feb 2019.

[4] B. Couraud, S. Norbu, M. Andoni, V. Robu, H. Gharavi, and D. Flynn, "Optimal residential battery scheduling with asset lifespan consideration," in *2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, pp. 630–634, Oct 2020.

[5] E. Cuisinier, A. RUBY, C. BOURASSEAU, P. Lemaire, and B. Penz, "Rolling horizon optimization: new approaches to balance short-term and long-term decisions for energy planning," in *ROADEF 2021 - 22ème congrès de la société Française de Recherche Opérationnelle et d'Aide à la Décision*, (Mulhouse, France), Apr. 2021.

[6] M. P. Marietta, M. Graells, and J. M. Guerrero, "A rolling horizon rescheduling strategy for flexible energy in a microgrid," in *2014 IEEE International Energy Conference (ENERGYCON)*, pp. 1297–1303, May 2014.

[7] J. Silvente, G. M. Kopanos, E. N. Pistikopoulos, and A. Espuña, "A rolling horizon optimization framework for the simultaneous energy supply and demand planning in microgrids," *Applied Energy*, vol. 155, pp. 485–501, 2015.

[8] AEMO, "Electricity demand forecasting methodology information paper," tech. rep., Australian Energy Market Operator Limited, Aug. 2020.

[9] S. G. Baratsas, A. M. Niziolek, O. Onel, L. R. Matthews, C. A. Floudas, D. R. Hallermann, S. M. Sorescu, and E. N. Pistikopoulos, "A framework to predict the price of energy for the end-users with applications to monetary and energy policies," *Nature communications*, vol. 12, pp. 18–18, 01 2021.

[10] A. Maheshwari, N. G. Paterakis, M. Santarelli, and M. Gibescu, "Optimizing the operation of energy storage using a non-linear lithium-ion battery degradation model," *Applied Energy*, vol. 261, p. 114360, 2020.

[11] C. O. Adika and L. Wang, "Smart charging and appliance scheduling approaches to demand side management," *International Journal of Electrical Power & Energy Systems*, vol. 57, pp. 232 – 240, 2014.

[12] H. Pandžić and V. Bobanac, "An accurate charging model of battery energy storage," *IEEE Transactions on Power Systems*, vol. 34, pp. 1416–1426, March 2019.

[13] A. Gonzalez-Castellanos, D. Pozo, and A. Bischi, "Detailed li-ion battery characterization model for economic operation," *International Journal of Electrical Power & Energy Systems*, vol. 116, p. 105561, 2020.

[14] P. J. Stuckey, K. Marriott, and G. Tack, *The MiniZinc Handbook 2.2.3*. Monash University and Data61 CSIRO, Oct. 2018.