Optimisation of residential battery integrated photovoltaics system: analyses and new machine learning methods

Rui Tang

A thesis submitted to fulfil requirements for the degree of
Doctor of Philosophy

Faculty of Engineering
School of Electrical and Information Engineering
The University of Sydney
2021

Abstract

Modelling and optimisation of battery integrated photovoltaics (PV) systems require a certain amount of high-quality input PV and load data. Despite the recent rollouts of smart meters, the amount of accessible proprietary load and PV data is still limited.

This thesis addresses this data shortage issue by performing data analyses and proposing novel data extrapolation, interpolation, and synthesis models. First, a sensitivity analysis is conducted to investigate the impacts of applying PV and load data with various temporal resolutions in PV-battery optimisation models. The explored data granularities range from 5-second to hourly, and the analysis indicates 5-minute to be the most suitable for the proprietary data, achieving a good balance between accuracy and computational cost. A data extrapolation model is then proposed using net meter data clustering, which can extrapolate a month of 5-minute net/gross meter data to a year of data. This thesis also develops two generative adversarial networks (GANs) based models: a deep convolutional generative adversarial network (DCGAN) model which can generate PV and load power from random noises; a super resolution generative adversarial network (SRGAN) model which synthetically interpolates 5-minute load and PV power data from 30-minute/hourly data.

All the developed approaches have been validated using a large amount of real-time residential PV and load data and a battery size optimisation model as the end-use application of the extrapolated, interpolated, and synthetic datasets. The results indicate that these models lead to optimisation results with a satisfactory level of accuracy, and at the same time, outperform other comparative approaches. These newly proposed approaches can potentially assist researchers, end-users, installers and utilities with their battery sizing and scheduling optimisation analyses, with no/minimal requirements on the granularity and amount of the available input data.

Statement of Originality

I certify that the intellectual content of this thesis is the product of my own work and that all the assistance received in preparing this thesis and sources have been acknowledged.

Signature:

Rui Tang

Name: Rui Tang

Authorship Attribution Statement

Chapter 3 of this thesis is published as [Tang, Rui, et al. "Impacts of temporal resolution and system efficiency on PV battery system optimisation." Proc. AsiaPac. Solar Res. Conf. 2017. Available at: http://apvi.org.au/solar-research-conference/wp-content/uploads/2017/12/029_R-Tang_DI_Paper_Peer-reviewed.pdf]. The conceptualisation, methodology, software and editing of the paper are done by the co-authors and me. I was in charge of the data curation, analysis, validation, visualisation and writing of the original draft.

Chapter 4 of this thesis is published as [Tang, Rui, et al. "Residential battery sizing model using net meter energy data clustering." Applied Energy 251 (2019): 113324. doi:10.1016/j.apenergy.2019.113324]. The conceptualisation, analysis and editing of the paper are done by the co-authors and me. I was in charge of the methodology, data curation, software, validation, visualisation and writing of the original draft.

Part of Chapter 5 of this thesis is published as [Tang, Rui, et al. "Generating residential PV production and electricity consumption scenarios via generative adversarial networks." Proc. AsiaPac. Solar Res. Conf. 2018. Available at: http://apvi.org.au/solar-research-conference/wp-content/uploads/2019/01/068_DI_Tang_R_2018.pdf]. The conceptualisation, validation and editing of the paper are done by the co-authors and me. I was in charge of the methodology, data curation, software, analysis, visualisation and writing of the original draft.

Part of Chapter 5 of this thesis is under review for a publication in Applied Energy. The conceptualisation, validation and editing of the paper are done by the co-authors and me. I was in charge of the methodology, data curation, software, analysis, visualisation and writing of the original draft.

Chapter 6 of this thesis is published as [Tang, Rui, et al. "Interpolating high granularity solar generation and load consumption data using super resolution generative adversarial network." Applied Energy 299 (2021): 117297. doi:10.1016/j.apenergy.2021.117297]. The conceptualisation, validation and editing of the paper are done by the co-authors and me. I was in charge of the methodology, data curation, software, analysis, visualisation and writing of the original draft.

As supervisor for the candidature upon which this thesis is based, I can confirm that the authorship attribution statements above are correct.
Supervisor Name: Philip H.W. Leong
Signature:
Date:

Acknowledgements

First, I want to express my deepest gratitude to my supervisor Prof. Philip H.W. Leong, who provides unwavering guidance and support throughout each stage of my study.

I'm also very grateful to my co-supervisors, Prof. Anthony Vassallo, Prof. Jin Ma and Dr. Jonathon Dore, who constantly offer valuable suggestions and knowledge to improve my research.

Many thanks to Dr. Farzad Noorian and Dr. Khalid Abdulla, for their extensive help and support during the early stage of my PhD. It was also a great pleasure to work with Dr. Baran Yildiz, who provided an incredible amount of constructive criticism and advice throughout our collaborations.

I also want to extend my deepest gratitude to my lab-mates, friends and colleagues. The completion of my dissertation would not have been possible without their joy, support and encouragement, especially during my difficult times.

I'm deeply indebted to my parents and my family, who are always there to offer support and encouragement. I would not have achieved anything without them.

Contents

List of Figures

List of Tables

Abbreviations

PV Photovoltaics. 1

DG Distribution generation. 4

LP Linear programming. 6

DCGAN Deep convolutional Generative Adversarial Networks. 6

SRGAN Super resolution Generative Adversarial Networks. 6

CHP Combined production of head and power. 9

ROI Return of investment. 11

LCOE Levelised cost of electricity. 11

DSM Demand-side management. 11

NPV Net present value. 12

MILP Mixed integer linear programming. 13

MINLP Mixed integer non-linear programming. 14

DP Dynamic programming. 14

MDP Markov decision process. 14

RL Reinforcement learning. 14

MPC Model predictive control. 14

SMPC Stochastic model predictive control. 14

SOC State of charge. 16

SSR Self-sufficiency rate. 18

CVI Clustering validity indicator. 19

CDI Clustering dispersion indicator. 19

DBI Davies-Bouldin index. 19

MIA Mean index adequacy. 19

SI Scatter index. 19

GHI Ground horizontal irradiance. 20

ARMA Autoregressive moving average. 20

CDA Conditional demand analysis. 20

GANs Generative adversarial networks. 21

WGAN Wasserstein generative adversarial networks. 21

WGAN-GP Wasserstein generative adversarial networks with gradient penalty. 21

ACGAN Auxiliary classifier generative adversarial network. 21

API Application programming interface. 25

RB Rule-based. 26

ToU Time-of-use. 29

LR Linear regression. 45

RF Random forest. 45

Lasso Least absolute shrinkage and selection operator. 45

CV Cross validation. 46

NRMSE Normalised root mean square error. 55

r2 r-squared value. 57

Chapter 1

Introduction

By the end of 2019, annual global photovoltaics (PV) installations reached 121 GW, bringing the total installed solar capacity to around 633 GW [1]. A decade ago, as shown in Figure 1.1, the total capacity was only 22.7 GW and this significant growth is mainly driven by the reduction in technology and installation costs and government schemes that have been supportive of the PV uptake. On the other hand, as of the end of 2019, PV generation only contributes 2.6% of the global power, indicating that PV still has much potential to be utilised [2]. Among the biggest problems facing the future uptake of PV are the grid integration challenges. As the penetration of distributed PV systems increases, it creates major issues for the grids which were originally designed for unidirectional energy flow. Another challenge to the PV uptake is the expiration of the generous feed-in tariffs, which have accelerated the adoption of PV in various countries and regions. As a result, solar feed-in tariffs are now lower than the general import tariffs which affects the financial returns of PV installations.

A net metering scheme is considered a feasible option to reduce the electricity costs for PV consumers. Different to gross meters where all the solar generation is exported to the grid, as illustrated in Figure 1.2, solar generation of customers with net metering schemes is first used on-site and the excess energy is then exported. As a result, the net metering scheme has been adopted in many different countries such as Australia [3], most states in the USA [4] and Germany [5] etc.

Energy storage is another viable solution to bring value to the PV consumers and utilities: not only because it increases the self-consumption of solar and thus reduces electricity costs for commercial and residential consumers, but also it can assist the grid by reducing the peak demand and avoiding losses associated to PV curtailment in regions with strict export limitations. Despite all the potential benefits that an energy storage system could offer, the penetration of storage systems is still low, mainly due to the high upfront costs of installing a battery system [6]. Besides, installing an energy storage system would also require a purchase of a multimode inverter to replace or add on top of the

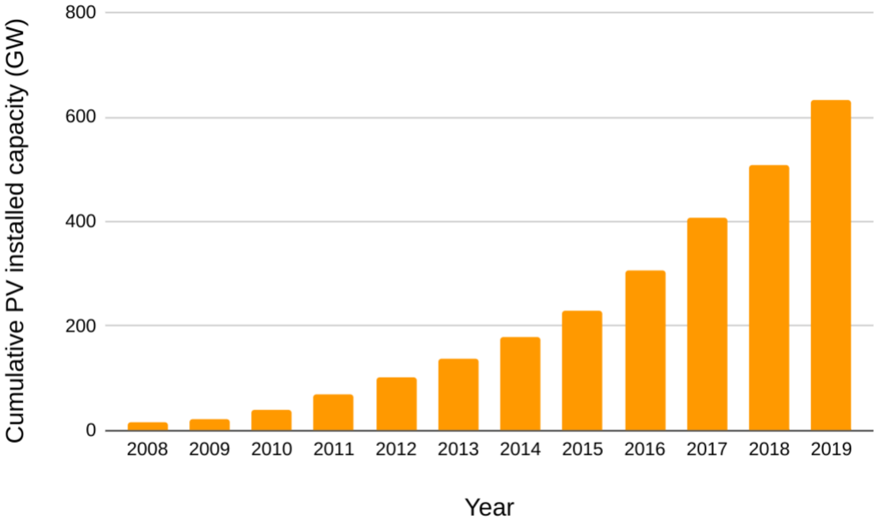

Figure 1.1: Global cumulative PV installed capacity 2008-2019 [2].

most commonly used current grid-connected inverters, which further increases the upfront investments [7]. An example system configuration of a PV battery system is shown in Figure 1.3, where a multimode inverter and a battery are retrofitted onto an existing PV system with minimal rewiring [7]. To fully realise the potential and maximise the financial returns of PV coupled battery systems for consumers and energy service providers, charging and discharging operations need to be optimised through effective planning and scheduling.

Solar generation is intermittent due to its dependence on weather and location. Furthermore, household electricity load is highly stochastic due to occupancy behaviours and socioeconomic factors. Optimisation of PV integrated battery systems, therefore needs to consider the uncertainties brought by PV generation and load. These can be modelled using a large amount of monitored data collected through monitoring devices and smart meters.

As a component of the Advanced Metering Infrastructure (AMI), smart meters can monitor and transfer data more frequently and efficiently compared to traditional interval meters [9]. Moreover, with two-way communication between the consumers and utilities, smart metering technology enables and enhances capabilities such as demand response programs, generation and consumption forecasting, optimisation of battery integrated distributed generation and time-variant tariffs [9]. Although some countries have started the rollouts of smart meters and the smart meter penetration is expected to increase the amount of accessible data, most consumers may not be able to properly access their data due to a lack of user-friendly tools and regulatory issues behind the data ownership [10]. Moreover, for third parties such as installers and researchers,

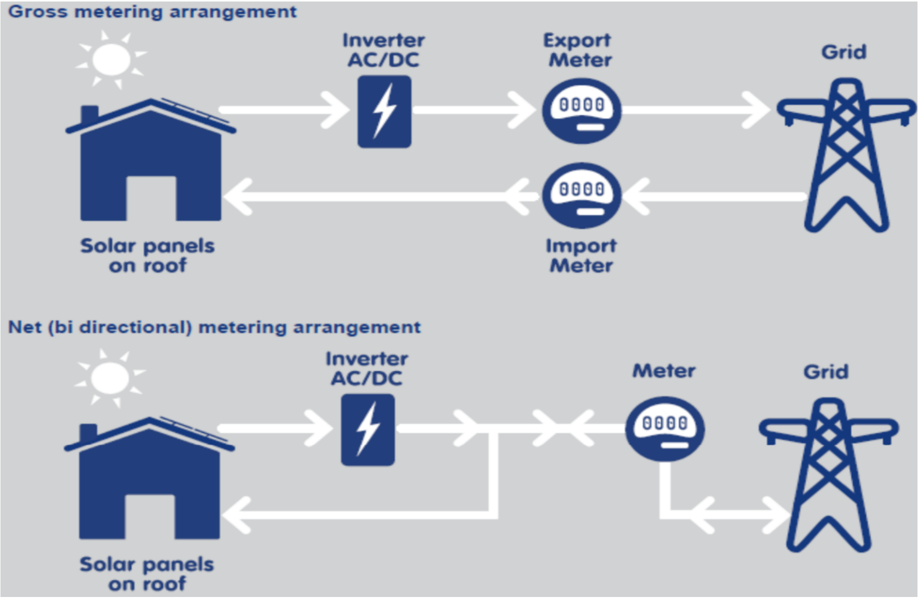

Figure 1.2: Gross metering scheme vs net metering scheme [8].

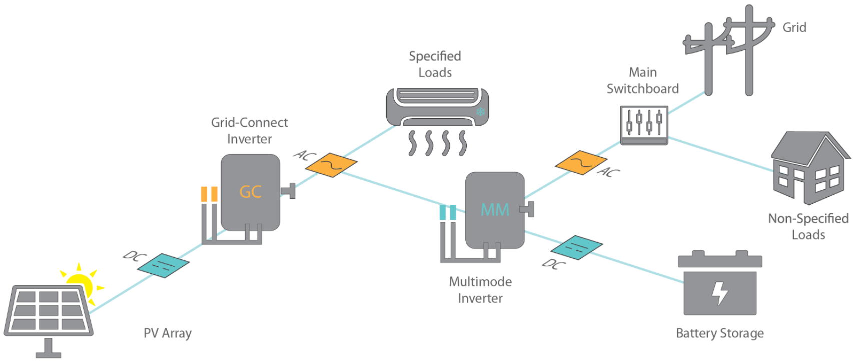

Figure 1.3: System configuration of a battery integrated PV system with two inverters [7].

accessing household meter data is still difficult due to privacy concerns, and the available public datasets are usually outdated and limited. Furthermore, currently, smart meter data's common temporal resolutions are still 15-minute or coarser [11] and most of the open access smart meter datasets are at 30 minute or hourly temporal resolution [12].

It remains unclear which temporal precision is best suited for PV battery optimisation models in terms of accuracy and computational costs. Moreover, Most studies in power scheduling optimisation of PV and battery systems tend to assume the battery conversion loss is linear to the energy flows of a battery. Hence, there is a need to investigate the data from real PV battery systems to assess the impacts of assuming a constant battery charging efficiency.

This thesis provides an analysis of the impacts of data granularity and battery system efficiency on the results of battery integrated PV optimisation models. Moreover, it develops practical and effective data extrapolation and synthesis models to address the issue of limited proprietary data for optimisations of battery integrated PV systems.

1.1 Research Motivations

For most optimisation studies related to PV battery power scheduling, the granularity of input PV and consumption data will determine the temporal resolution of an optimisation cost function. This is because most models have a single fixed-length horizon with constant resolution and formulation of the optimisation problem is more straight-forward when the input data to an optimisation model has the same granularity as the output control signals. The use of relatively low temporal resolution may lead to errors in estimated costs as realised costs are derived instantaneously in a real-time scenario. The literature remains unclear how low temporal precision could impact the optimised costs of objective functions in a PV battery power scheduling model. Moreover, a resolution finer than 1 minute has not yet been explored by distributed generation (DG) optimisation analysis. There is no published research looking at evaluating the impacts of various battery efficiency settings in a PV battery optimisation model to the best of the author's knowledge.

Despite many exciting signs of progress made in the recent PV battery power scheduling and sizing optimisation literature, the practicability of these approaches remains questionable for the following reasons:

(1) To build a robust model, a minimum amount and high-quality of input PV / weather and load data are often required whereas in practice, such data might not be available.

(2) Very little research considers the limited proprietary data problem. A residential household may have insufficient measured data for these battery size determination/power scheduling optimisation models to work properly.

(3) One potential solution for the above issues is to perform data extrapolation using a customer's consumption and generation patterns extracted from the limited historical data. However, to the best of the author's knowledge, so far there is minimal work related to power data extrapolations that can produce a sufficient amount of input data for PV integrated battery system optimisations.

In the absence of any measured historical data, data extrapolation methods are not applicable and synthetically generated data can be used to model the data distributions and generate possible trajectories of PV and load power. Previous researchers have explored synthetic PV and household electricity load data generation, yet the practicability and reliability of these approaches are still questionable for the following reasons:

(1) Some studies apply probabilistic models which rely on specific statistical assumptions that may affect the effectiveness of data synthesis (e.g. Markov property is assumed for Markov chain based generative models [13, 14]. This refers to the memoryless property of a Markov process and specifies that the conditional distribution of future events in a Markov process depends only on the current event [15]). It is reported in [16] that even though the synthetic load data generated by Markov chains shows similar statistical characteristics to real measured data (e.g. mean, standard deviation), the captured temporal correlations are not adequate.

(2) Some approaches [17, 18] require additional survey and weather data as model inputs, which in practice can be even harder to obtain compared to meter data. For instance, national time-use surveys are rarely conducted [19] and often outdated and accurate solar irradiance data requires either an on-site irradiance sensor or data access from a third-party organisation.

(3) The validations of synthetic data generation models are often conducted on minimal real-time data [20, 21].

To date, most smart meter datasets still have data granularities of 15-minute or coarser. While this granularity level could be sufficient for billing or deriving the aggregated generation or consumption pattern, it may not fully capture the weather transients or consumption spikes. One potential solution to address the above-mentioned issues is to synthetically interpolate higher resolution smart meter data from lower resolution data. However, in the existing literature, very few studies have looked into this topic.

1.2 Research Contributions

Motivated by these facts, this thesis addresses the issue of limited proprietary data for PV battery optimisations using developed analyses, data extrapolation and synthesis models. An overview of the developed analyses and approaches is shown in Figure 1.4. A sensitivity analysis is carried out to investigate the impacts of temporal resolution and battery efficiency settings on the optimised costs of PV battery scheduling models, where it is concluded that 5-minute temporal resolution achieves a good balance between accuracy and computational costs. Then based on the scenarios of limited proprietary data, different approaches are developed to generate sufficient 5-minute PV & load data through data extrapolation, synthesis and interpolation. After that the produced datasets are fed into a battery size optimisation model to validate the effectiveness of these models.

Overall, the contributions of this thesis are to:

1. Carry out the first sensitivity analysis to investigate the errors related to various granularities in optimised costs of a rule-based battery scheduling algorithm and a linear programming (LP) optimisation model. This is also the first DG optimisation analysis that evaluates a temporal resolution finer than 1-minute.

2. Perform a clustering analysis on residential PV customers using net meter energy data. To the author's knowledge, this is the first work performing a clustering analysis on net meter energy data. Hopefully, with the ongoing worldwide adoption of net meters, this work could illustrate a new direction to load clustering research as gross load data will no longer be collected from net meter customers. The net meter clustering approach is then used to build a data extrapolation model, for the first time addressing the insufficient net meter data problem in battery size optimisation.

3. Present a single Deep Convolutional Generative Adversarial Networks (DCGAN) model that can simultaneously generate synthetic residential gross/net meter solar and load data, taking account of the correlations between on-site PV and load power. Previous approaches used independent models [22].

4. Propose the first work which is a Super Resolution Generative Adversarial Networks (SRGAN) based model to interpolate 5-minute average PV generation and load power data from 30-minute/hourly average PV and load power measurements. The synthetically interpolated high temporal precision power data is validated in a PV integrated battery optimisation model, which for the first time, addresses the issue of applying coarse PV and load data in modelling residential PV battery systems.

1.3 Thesis Layout

The remaining parts of this thesis are organised as follows: Chapter 2 introduces the relevant literature for PV integrated battery system sizing and power scheduling optimisation, load clustering and synthetic power data generation. An analysis of the impacts of data granularity and battery efficiency settings on PV integrated battery system optimisation is given in Chapter 3. Chapter

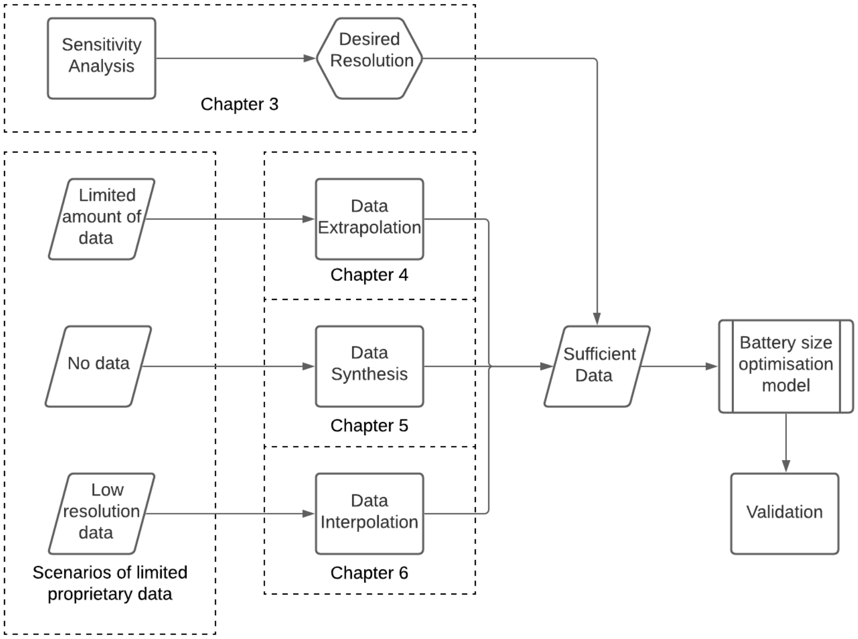

Figure 1.4: Overview of the proposed analyses, approaches and their corresponding chapters

4 introduces a battery sizing model which is robust to limited amounts of input data. A DCGAN based model which can generate high-quality 5-minute synthetic residential PV and load power data is presented in Chapter 5. The proposed model is also used in a battery simulation model, to estimate electricity costs and perform energy storage sizing for new residential customers with no historical data. Chapter 6 includes a SRGAN model which creates 5-minute data from 30-minute and hourly PV and load data. Chapter 7 summarises and concludes the thesis.

Chapter 2

Literature Review

2.1 Introduction

An overview of the topics covered in this chapter is shown in Figure 2.1. Distributed generation (DG) and types of battery integrated DG optimisation are first introduced. Then this chapter focuses on past studies of battery integrated PV power scheduling optimisation and planning optimisation, with their adopted optimisation algorithms, optimisation objectives, types of input data, optimisation horizons and storage efficiency settings summarised. It should be noted that demand-side management optimisation is not within the scope of this thesis, therefore the studies in this area are not reviewed in this chapter. After that, as the models described in Chapter 4 and Chapter 6 both apply load clustering, the related literature of load clustering is discussed. Lastly, the relevant studies of synthesising PV and load power data are summarised.

2.2 Optimisation of Battery integrated PV Systems

2.2.1 Distributed Generation

Distributed Generation, commonly defined as a type of energy resource connected directly to a distribution network or on the network's customer side [23], has recently become more competitive in the electricity market. DG potentially has the advantages of lower environmental pollution, reduced power loss and less required transmission capacity [24] and it can be categorised into three main groups regarding to generation technology: renewable energy resources (wind, solar PV, biomass etc.), modular generating systems (diesel generators, microturbines, fuel cells) and combined production of heat and power (CHP) ([23, 25]). Due to the natural intermittency of some of the DG resources (such as wind and PV), energy storage systems are often integrated with DG systems to

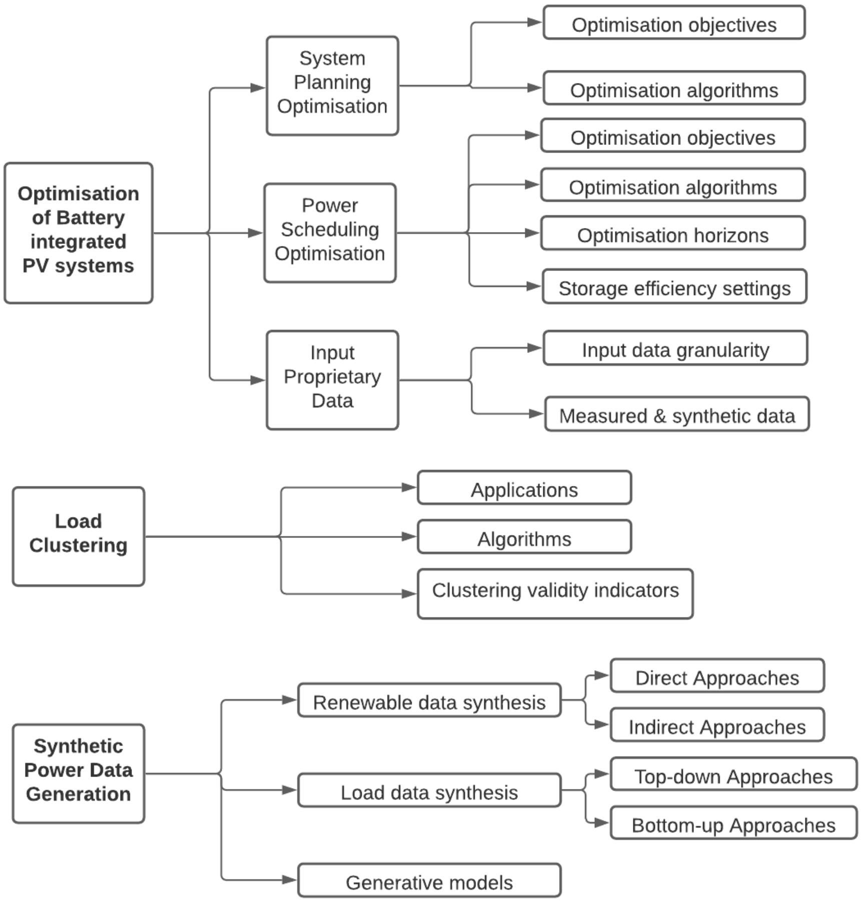

Figure 2.1: An overview of the reviewed topics.

reduce the mismatch between demand and generation and hence increases the overall benefits of DG systems.

2.2.2 Varieties of Optimisation for Battery integrated Distributed Generation

Energy storage systems generally require a large amount of upfront cost therefore the designing and implementation processes of optimisation strategies are crucial to the return of investment (ROI) of storage integrated DG systems. In the existing literature, optimisation strategies focused on various aspects of battery integrated DG systems have been well explored by many researchers. Table 2.1 illustrates reviewed studies in this area, they can be classified into three main categories.

Table 2.1: Optimisation type of reviewed studies on storage integrated DG.

Types of Optimisations	Subcategory	Reviewed Studies
System Planning Optimisation		[26–29]
Demand-side Management Optimisation		[30–37]
Power Scheduling Optimisation	Renewable Energy System Power Scheduling Optimisation	[24, 25, 29, 34, 38–74]
	Microgrid Power Scheduling Optimisation	[75–90]

System Planning Optimisation

Some studies have investigated the optimisation of system planning for battery integrated DG systems. System planning models for battery integrated PV systems were proposed by [26], [27] and [28] which derived the optimal system capacity to maximise the system finacial returns. Another approach proposed by [29] calculates optimal battery size for a stand-alone hybrid system to minimise the levelised cost of electricity (LCOE).

Demand-side Management Optimisation

Optimisation through demand-side management (DSM) has also been studied for storage integrated DG systems. Controllable loads in a storage coupled DG system are scheduled for various optimisation objectives such as peak shaving, minimising electricity/fuel cost and increasing self-consumption (reviewed studies are listed in Table 2.1).

Power Scheduling Optimisation

Power scheduling optimisation for storage integrated DG systems has gained increased attention in the last decade, it can be roughly classified into two categories based on their system setups. The first category is storage integrated renewable energy system power scheduling optimisation. It should be noted that a noticeable amount of them are mainly focused on power scheduling of PV integrated battery systems [25, 29, 34, 49–74]. The second group of studies investigates the power scheduling optimisation of microgrids which is different from renewable energy power scheduling as the system configurations also consider modular generating systems and CHP.

2.2.3 Battery integrated PV System Planning Optimisation

The main difference between system planning optimisation and power scheduling optimisation is that the former focuses on size optimisation of PV or/and battery. In contrast, the latter often aims to optimise the battery charging & discharging activities.

Optimisation Objectives

Many studies perform the techno-economic analysis of PV-integrated battery systems, where many focus on the battery size determinations. These studies have adopted various economic, technical and environmental indicators to be optimised in their modelling approaches [91]. The economic criteria includes levelised cost of electricity (LCOE) [92–94], net present value (NPV) [95–97], ROI [98] and cost-competitiveness with the grid import rate [99]. The adopted technical indicators consist of voltage deviations [94, 100], energy losses [100] and frequency control [101]. Environmental criteria is generally related to CO2 emissions where levelised CO2 equivalent life cycle emissions and damage cost of CO2 emissions were respectively considered in [94] and [102].

Optimisation Algorithms

Different optimisation algorithms have been adopted to find the optimal system configuration. Most of them are also used for power scheduling optimisations, as mentioned earlier. In [103], MILP was adopted to calculate the lower and upper bounds of the optimal battery sizes for a grid-connected solar system where the electricity costs stay the same when battery size exceeds the upper limit and increase significantly when the storage size is below the lower limit. MILP was also applied in a similar manner in [92, 96] which optimises the system configuration and operation schedule of a PV integrated battery system. Exhaustive search was adopted in [104] to look for the battery system configuration with the lowest LCOE. A similar approach using exhaustive search was performed in [93], however it was proposed for battery sizing in off-grid renewable energy systems and optimising battery control strategies. Stochastic MINLP was applied in [105] to optimise sizes and power schedules of a PV integrated battery system, with a Monte Carlo approach to model the uncertainties in PV production. A GA-based approach [100] was applied to optimise the sizes and locations of battery-coupled distributed PV generators in distribution networks. GA was also applied in [102] to transform the optimisation cost function into a linear programming (LP) function. Then the LP function is solved to find the optimal placements, sizes and power schedules in a distribution network. Authors in [97] applied a dynamic programming approach to optimise sizes and energy dispatches in lithium-ion battery integrated commercial PV systems.

2.2.4 Battery integrated PV System Power Scheduling Optimisation

Optimisation Objectives

Different studies on power scheduling of the storage-coupled PV system have applied various optimisation objectives. These objectives define their individual objective function, which is the core part of an optimisation model.

A significant part of the research seeks to find power scheduling optimisation models to minimise electricity cost [34, 49, 50, 53, 54, 56, 57, 59, 62, 65, 66, 71, 74, 81, 88]. Another well-adopted optimisation objective is peak demand shaving which could lead to benefits to both grid and consumers [25, 50–52, 55, 61, 69]. Some studies considered models to find optimal scheduling strategies on maximising the lifetime ROI of a battery integrated solar system [29, 63]. Authors in [70, 106] proposed optimisation models to maximise self-consumption of battery integrated renewable energy systems. The cost associated with battery degradation was also considered one of the optimisation objectives in some studies [49, 54, 76]. Authors in [65, 66] also adopted an optimisation objective to mitigate over-voltage issues caused by reverse power flows of PV. Optimisation model proposed by [85] included increasing grid energy security as one of their optimisation objectives. Moreover, the authors in [72] introduced a power scheduling optimisation model to minimise the line loss of PV battery systems.

Optimisation Algorithms

Various optimisation algorithms have been applied to solve the scheduling optimisation objective functions for battery integrated PV systems.

Rule-based algorithms have been considered and applied for several studies [51, 52, 59, 63, 70, 71, 76, 79, 87] due to its advantages of simplicity and high flexibility for implementation. It has also been used as a base case for comparing with other more advanced optimisation models.

Linear programming (LP) is generally defined as maximising or minimising a linear function by applying linear inequality or equality constraints [107]. It has been applied by some studies [25, 34, 50, 55, 61, 68, 74, 78] as it can converge at a low computational cost and can guarantee the solution is optimal if the optimisation problem is linear.

When some of the LP optimisation variables are restricted to discrete integers, the optimisation problem becomes a Mixed Integer Linear Programming (MILP) problem. A MILP problem formulation has been adopted by several power scheduling studies on battery coupled PV systems [54, 56, 77, 81, 82, 88, 89, 106].

Suppose an optimisation cost function is non-linear and has integer variables. In the case, the problem becomes a Mixed Integer non-linear Programming (MINLP) problem which was used in energy scheduling optimisation for microgrids [84].

Like LP, Quadratic Programming (QP) uses linear constraints but its objective function is quadratic convex [108]. Models using QP were applied in a few studies for power scheduling of battery integrated PV systems [66–68].

Genetic Algorithm (GA), a type of evolutionary computation technique, has also been applied for storage scheduling problems with PV [29, 60]. GA can solve optimisation problems by constantly modifying a group of solutions towards an optimal solution. It can solve stochastic and non-linear optimisation problems that LP, QP and MILP can not easily solve. Another evolutionary optimisation technique, Particle Swarm optimisation (PSO), was also applied for power scheduling PV battery systems [75]. The advantage is PSO is that it requires fewer adjustments on parameters and easier to implement [109].

Dynamic Programming (DP) solves optimisation problems by dividing a complex problem into dependent sub-problems and then utilising the solutions of these simpler sub-problems to find an optimal global solution [110]. DP assumes the model environment is a Markov decision process (MDP) and has been favoured by a noticeable amount of researchers in this area [25, 49, 50, 53, 56–58, 65, 66, 69, 76, 86].

Reinforcement Learning (RL) is a machine learning technique which allows a software agent to learn optimal behaviours using the feedback from the environment. RL has quite similar working principles to DP however the major difference is that DP algorithms assume perfect knowledge of the model and transition probabilities, whereas RL only needs access to a set of samples without knowing exact information of the environment model. It has been applied for storage coupled microgrid scheduling by [80, 83].

Model Predictive Control (MPC), also referred as Receding Horizon Control (RHC), is a control design technique that can be applied for scheduling optimisations. Instead of applying a pre-computed control sequence, MPC determines the current control, at each optimisation period, by numerically solving an open-loop optimisation problem within the finite prediction horizon using the current system state as the initial state [111]. Stochastic MPC (SMPC) introduces the probabilistic descriptions of uncertainties in MPC into a stochastic optimisation problem [112]. As forecasts of demand and generation often have their intrinsic errors, MPC and SMPC have the potential of taking these errors into account in the following optimisation iteration and hence increase the optimisation robustness. Several studies have applied MPC models and they are generally coupled with algorithms discussed above, such as RB, LP, DP and MILP [50, 59, 62, 73, 85, 89, 90].

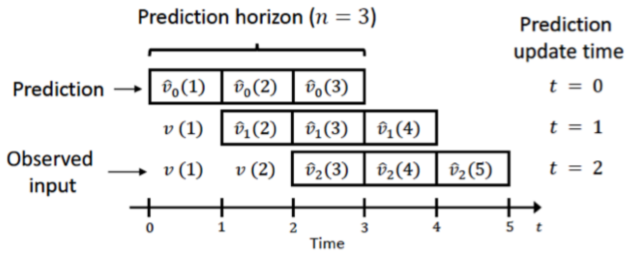
Prediction horizon ($n = 3$)
Prediction update time
Prediction
$\hat{v}_0(1)$ $\hat{v}_0(2)$ $\hat{v}_0(3)$
$t = 0$
v (1) $\hat{v}_1(2)$ $\hat{v}_1(3)$ $\hat{v}_1(4)$
$t = 1$
Observed input
v (1) v (2) $\hat{v}_2(3)$ $\hat{v}_2(4)$ $\hat{v}_2(5)$
$t = 2$
0 1 2 3 4 5 t
Time

Figure 2.2: An example of a MPC framework with a 5-step optimisation horizon [113].

Optimisation Horizon

Optimisation horizon for a sequential decision-making process is referred to as the control horizon considered in the optimisation model. Theoretically, the optimisation horizon of a PV battery system control problem is the lifetime of the system however, this is rarely adopted in the previous studies for a couple of reasons: 1. Long horizon exponentially increases the computational costs for optimisation algorithms such as LP and DP. 2. For optimisation models that require PV and load forecasts, it is not feasible to get forecasted data with adequate accuracy for a horizon equal to the lifetime of a system.

Most of the reviewed studies adopted 24-hour optimisation horizons [24, 25, 29, 34, 38, 39, 41–43, 46–51, 55, 58–62, 65–72, 74, 75, 79–81, 84–86, 88, 89, 106]. A 12-hour optimisation horizon was adopted by a few proposed optimisation models [40, 64, 73]. Optimisation horizons of 48-hour and 72-hour were respectively applied for optimisation models proposed in [57] and [78].

Due to the intrinsic intermittency of PV and high variability in residential and commercial electricity consumption profiles, a significant number of studies incorporated the forecasts of PV generation and consumption to enhance the effectiveness of their power scheduling models [29, 34, 49–53, 55–59, 61–64, 66–71, 73, 75, 76, 78, 79, 85, 89, 106].

For optimisation models that require forecasting PV and consumption, another term is generally referred to as the prediction horizon, which is simply the look-ahead horizon for forecasted PV and load. In a noticeable amount of studies, the length of the prediction horizon is equivalent to the control horizon as control signals in the optimisation horizon are dependent on the predictions of PV and consumption. On the other hand, for a MPC approach illustrated in Figure 2.2, the prediction horizon is not necessarily the same as the optimisation since MPC will update its forecasts before the optimisation horizon ends. Furthermore, several studies adopted a multi-stage approach in their optimisation frameworks. This technique first solves the optimisation problem on an extended horizon with a relatively low-resolution. It then combines the derived low-resolution control signals with control decisions computed by an optimisation horizon with shorter length but finer resolution [53, 56, 78].

Storage Efficiency Settings

Storage efficiency plays a vital role in the system setup of a PV battery power scheduling optimisation problem as not only it affects the system efficiency, but it can also influence the State of Charge (SOC) constraints in the optimisation formulation. Most studies in the power scheduling optimisation of PV and battery systems tend to assume the battery conversion loss is linear to the energy flows of a battery. Battery efficiency settings used by reviewed literature can be categorised into three main types, as shown in Table 2.2.

Table 2.2: Battery Efficiency Settings applied in existing PV battery scheduling optimisation studies

Storage Efficiency Setting	References
Perfect Conversion Efficiency	[38, 48, 61, 64, 67, 68, 71, 80, 81, 86, 88, 90, 106]
Constant Charging/ Discharging Efficiency	[24, 25, 34, 39–44, 47, 49–55, 58–60, 63, 65, 66, 70, 72–79, 84, 85, 89]
Efficiency derived from quadratic curves	[46, 56, 57, 69]

A noticeable number of studies assume perfect battery conversions, in other words, the efficiency is assumed to be 100% and no energy is lost during charging & discharging activities. A significant amount of studies incorporates a constant charging/discharging efficiency, where the battery efficiency is constant regardless of the battery charging/discharging power. Several studies have adopted a quadratic battery efficiency curve where the battery charging/discharging efficiency is dependent on the input/output power.

2.2.5 Input Proprietary Data

Input Data Granularity

Different PV and load datasets have been applied in various reviewed power scheduling studies. The temporal resolutions of these datasets are strongly dependent on the adopted sampling rates of electricity meters and weather stations, along with forecasting and modelling techniques of load consumption and PV generation.

The temporal resolution of an optimisation model for a sequential decision-making problem is generally defined as how frequent a decision is implemented.

For most studies related to PV battery power scheduling, the granularity of input PV and consumption data determines the temporal resolution of the optimisation model. This is simply because most proposed optimisation models have a single fixed resolution and horizon and formulation of the optimisation problem would be much simpler when the input data has the same granularity as the output control signals.

1-minute resolution measured PV and consumption data was used in a microgrid scheduling optimisation study to evaluate the performance of a DP optimisation framework [86]. A couple of studies have used datasets with resolutions close to 5-minute: forecasted 4-minute and 5-minute PV and consumption datasets were included in the optimisation models proposed by [73, 79]. In the optimisation frameworks developed by [51, 69], 10-minute PV and demand datasets were applied to evaluate their models. 15-minute PV and consumption datasets have been used in several PV battery power scheduling studies [52, 55, 56, 58, 61, 63, 85]. Moreover, a noticeable amount of researches utilised 30-minute PV and load data in their PV battery scheduling optimisation models [49, 50, 57, 67, 68, 89]. A large number of studies applied hourly PV and load data in their optimisation studies [25, 29, 34, 53, 59, 60, 62, 64, 65, 72, 74, 75, 77, 81, 84, 88]. Table 2.3 summarises the temporal resolutions of PV and consumption data used in reviewed PV battery scheduling optimisation studies.

Table 2.3: Temporal resolution applied in existing PV battery scheduling optimisation studies.

Temporal Resolution	References
1-minute	[86]
4, 5-minute	[73, 79]
10-minute	[51, 69]
15-minute	[52, 55, 56, 58, 61, 63, 85]
30-minute	[49, 50, 57, 67, 68, 89]
hourly	[25, 29, 34, 53, 59, 60, 62, 64, 65, 72, 74, 75, 77, 81, 84, 88]

A few approaches in the literature have compared various granularities used in DG optimisations. Some explorations have been conducted to evaluate the impacts of applying input data with various resolutions. The impacts of temporal resolution on the optimisation results of micro combined hear and power (CHP) systems are analysed in [114], where noticeable differences in optimal capacity, carbon dioxide emission reduction and lifetime costs are found between using 1-hour and 5-minute load energy data. On the other hand, there were minor differences between the results derived from 5-minute and 10-minute resolution; hence the authors concluded that finer resolution may only lead to insignificant improvements and much more computational costs. An analysis was done to explore the effects of data granularity on the imports and exports of a DG system. The study concluded that low-resolution data leads to underestimations of imports and exports [115]. Authors in [44] investigated impacts of data resolution on the optimal sizes of components such as PV, wind, battery and diesel in a renewable system, they found the impacts are strongly related to system configuration, and it is difficult to make a simple granularity recommendation. A study completed by [116] explored effects of applying low-resolution data on the modelling of a residential PV battery system. The results showed that coarse load data could cause overestimations of battery lifetime and underestimations of a battery's contribution to PV self-consumption. The impacts of data granularity on DG capacity and estimated losses were investigated by [117], they recommended a resolution finer than 1 hour is not necessary as differences in results are negligible when using high-resolution data. Authors in [11] analysed the influence of PV and load data granularity on self-consumption and sizing of a PV battery system. They found temporal precision of load data is more critical to the estimation of self-consumption rate. For a system with a relatively low and stable demand profile, 15-minute data is sufficient for the determination of self-consumption rate, whereas 5-minute or finer temporal precision is required for the sizing of battery inverter power. Moreover, their study concluded that hourly resolution is sufficient for the sizing of PV battery systems. Authors in [50] demonstrated that the estimations of storage value can be influenced by the temporal resolution of input PV and load data for a residential PV storage system. An average of 17% difference was found between using 1-minute and 30-minute data for a simulated site configuration in which the battery is controlled by a rule-based algorithm designed to maximise self-consumption of PV.

Measured vs Synthetic Data

Despite all the exciting progress in battery optimisation models, most studies still use synthetic PV or consumption data. In [13], 20 years of solar irradiance data generated by discrete-time Markov chains was fed into a battery simulation model to assess the economic benefits of storage on reducing imbalance penalties. Another study [17] determined the capacity distributions of a battery-supercapacitor hybrid energy storage system in a micro-grid, applied probability density estimations and Monte Carlo simulations to generate synthetic input data of wind speed, irradiance and load. One exception, reference [95], where net meter energy data from 79 solar households was adopted. However, the dataset had a considerable amount of missing data (68 days out of one year).

A few studies have emphasised the importance of using real-time load data in the system planning optimisation of PV battery systems. Authors in [118] compared real-time and aggregated load profiles and concluded that adopting aggregated load data may result in overestimated self-consumption and underestimated total costs. Studies like [119] and [120] showed households with various consumption patterns may result in quite different end net present values (NPVs) and self-sufficiency rate (SSR) for battery integrated solar systems. The wide adoption of synthetic data is likely due to the lack of high-quality publicly available generation and consumption datasets. Moreover, even in practice, battery installers or utility which often have more direct contacts with solar customers, suffer from the insufficient data during their decision-making processes of battery system configuration.

Very few research considers the limited input data problem stated above. Authors in [121] used a techno-economic model to compute the optimal PV and battery configurations for various non-solar customers and use the simulation results to develop a machine learning model that could predict the optimal configuration, NPV and SSR using a limited amount of load data. Although this model has achieved promising results, it still requires weather data to generate synthetic PV data. Another possible factor that could affect the practicability of the model is that a single change in the techno-economic parameters would require re-simulating and re-training of all the households in the training set.

2.3 Load Clustering

Load clustering has been applied for many studies concerning the analysis of electricity consumption data. A couple of papers have given comprehensive reviews on the applications, techniques and evaluation metrics of clustering [9, 122]. Load profiling, which is generally referred as identification of typical consumption profiles over a certain period, is one of the main applications of load clustering and it can be used for a better understanding of consumer behaviours [123], tariff designs [124] and demand strategies [125]. Customer classification also uses load profile clustering to create cluster labels related to household characteristics and demographic information [126, 127]. Moreover, load clustering has also been applied to enhance the performance of load forecasting algorithms [128, 129].

A variety of load clustering techniques have been attempted in the literature, such as hierarchical clustering [128, 130], k-means [123, 131], fuzzy k-means [132], follow the leader [133], self-organizing map [134], support vector clustering [124] and probabilistic neural networks [135]. The number of clusters needs to be defined manually for non-hierarchical clustering models (e.g. k-means clustering), although this is not required for hierarchical and follow the leader models [9].

Various clustering validity indicators have been applied to evaluate the performances of clustering algorithms; most of them are defined using Euclidean distance metrics [122]. Commonly used clustering validity indicators (CVIs) include Clustering Dispersion Indicator (CDI) [124], Davies-Bouldin Index (DBI) [132], Mean Index Adequacy (MIA) [122], modified Dunn Index [136] and Scatter Index (SI) [124].

Clustering research taking account of consumers with on-site generation remains limited. Authors in [137] applied a self-organizing map clustering model on 300 Australian households with installed PV systems which reveals a self-consumption behaviour within gross meter solar customers. A case study was demonstrated in [138] which shows how clustered consumption profiles can be used for the size planning of a PV and energy storage system on a commercial building.

2.4 Synthetic Power Data Generation

2.4.1 Renewable Data Synthesis

Previous studies of synthetic power data generation (also referred as scenario generation) for renewable energy resources can be categorised as *indirect* and *direct* approaches. For indirect renewable power scenario generation, synthetic weather data such as solar irradiance and wind speed is generated first, then fed into a renewable power system model to output synthetic power scenarios. In [13] and [14], synthetic clearness index values were generated by a first order Markov Chain, then synthetic ground horizontal irradiance (GHI) data was calculated. The synthetic GHI was used to simulate PV generation through a PV system simulation model. Synthetic solar irradiance and wind speed data were respectively generated via estimated normal and weinull distributions in [17], which was then converted into synthetic system power outputs. A similar approach was performed in [139] and [140], where probability density estimations were performed to produce synthetic wind speed and power data. Autoregressive moving average (ARMA) models were developed in [141–143] to first characterise the autocorrelation within historical wind speed data, then produce synthetic wind speed scenarios. Direct power scenario generation models generally only use historical power data as inputs instead of weather-related data. An approach to generate synthetic nearby sites' solar power data was developed in [144], which randomly mixes measured PV ramping events of nearby solar systems to produce synthetic PV power values. Authors in [145, 146] adopted copula models which fits the historical wind/solar power data into a multivariate probability distribution to sample synthetic wind/solar data.

2.4.2 Load Data Synthesis

Studies that model electricity load data can be categorised into top-down and bottom-up approaches [147], where bottom-up models are often applied to generate synthetic load profiles. Bottom-up approaches model the electricity demand of a small group of households by utilising information such as historical electricity load and weather data, appliance characteristics and occupant behaviours [148]. Then the load of the whole targeted geographical area is determined by extrapolating the modelled load demand of the smaller representative group [147]. As a popular branch of bottom-up approaches, conditional demand analysis (CDA) was first developed in [149] and further improved in [150, 151]. In [149], household and appliance survey data and the corresponding monthly electricity load data were analysed, where a linear regression model was proposed to predict monthly and yearly electricity load given the appliance counts and interaction variables such as number of occupants, income and electricity price. A neural network based bottom-up model was developed in [152], where air temperature and solar irradiance were adopted as inputs of the neural network to predict the hourly load of a residential building in Greece. Some other so called *engineering* bottom-up approaches do not require statistical analysis of historical load data to develop their models. Instead, they function by directly utilising the characteristics of appliances and household building envelopes [148]. Although these models are more flexible and capable of incorporating new appliances and building types, it is more challenging to take occupancy behaviours into account. Authors in [153] reconstructed household load curves by applying behavioural and technological probability functions derived from appliance distributions, demographic and lifestyle data. A high-resolution bottom-up electricity demand model was developed in [18], where the UK 2000 Time Use Survey data [154] was adopted to infer the probabilities of occupants switching on different appliances throughout a day. Then by aggregating the appliance power consumptions using their power characteristics, synthetic load curves were created for a given household.

2.4.3 Generative Models

In the last decade, compared to the significant progress made in classification tasks by discriminative models which is another main type of machine learning approach, generative models have less of an impact on synthetic data generation tasks due to the difficulties in approximating many intractable probabilistic computations in maximum likelihood estimation and utilising the benefits of piecewise linear units in the generative context [155]. As a popular machine learning research field, Generative Adversarial Networks (GANs) [155] has some potential benefits over the above-mentioned data synthesis methods: (1) GANs are a type of data-driven generative model, which means only historical meter data is required; (2) Assumptions that are required by other models such as Monte Carlo approximations or Markov chains are not needed; (3) GANs can leverage deep convolutional neural networks to generate high-quality samples which have been very promising in the image synthesis field [156]. Despite these advantages, the use of GANs on generating synthetic solar and household load data has been minimal. A Wasserstein GAN (WGAN) based model was developed in [21] to generate synthetic 5-minute solar power data, where they adopt a one-year 5-minute dataset of 32 solar plants and use 80% of daily samples as training data. A day ahead weather classification model was proposed in [20] where a Wasserstein Generative Adversarial Network with gradient penalty (WGAN-GP) was used to perform data augmentation on 15-minute solar irradiance data collected by a weather station. Results showed that augmented data can improve the classification accuracies. In [157], An Auxiliary Classifier GAN (ACGAN) was developed to generate synthetic 30-minute weekly load profiles conditioned on load patterns obtained by k-means clustering, and the study used a load dataset of 500 households.

2.5 Summary

In summary, this chapter focused on past studies regarding optimisations of battery integrated PV systems, including three major aspects: system planning optimisation, power scheduling optimisation and input proprietary data.

As illustrated in Table 2.3, most studies in PV battery scheduling adopted low-resolution data. This is expected for several reasons: 1. Smart meters or monitoring systems are not widely adapted in households until the last few years, old interval meters generally have lower sampling rates; 2. To date, publicly accessible high-resolution PV and consumption datasets are still quite limited; 3. High granularity input data could lead to high computational costs.

Recent contributions in optimisation research of PV-integrated battery system can be grouped under one of the following categories: 1. New types of optimisation criteria [99]; 2. More thorough considerations of optimisation objectives [93, 94, 102]; 3. Optimisation of battery control strategy or scheduling added on top of configuration determination [92, 97, 98, 105]. 4. New battery applications in renewable energy systems [97, 100].

There are pretty limited studies looking into the limited proprietary data problem for battery-integrated PV system optimisations. Moreover, it is still unclear how the temporal resolution of the input PV & load data can affect the optimised costs and savings of a PV-battery optimisation model.

Clustering is often considered an effective tool to obtain valuable information about customer consumption behaviours and has drawn the attention of many researchers. However, to date, the applications of this technique have mainly focused only on the electricity consumption data, ignoring the solar generation data despite the significant growth of residential solar customers.

Generative Adversarial Networks has become a quite popular machine learning technique due to its effectiveness on data synthesis and it has been applied for generating synthetic smart meter data. However, it is found that these studies have certain limitations which may affect the potential success of GANs for this area of research:

(1) Often, small datasets were used (e.g.[20]). As individual residential households could have various PV and load profiles, more significant number of samples are necessary to capture their power data distributions adequately. One exception is [157], however the authors only selected four profiles for the validation of individual synthetic load curves and no information was given on how many profiles were used for model training and validating of aggregated synthetic profiles.

(2) Data synthesis was performed for low granularity data in [20] and [157], where some applications such as PV integrated battery system power optimisation may require 5-minute or higher temporal resolution for accurate estimations of battery savings and electricity costs [158].

(3) The metrics to validate the synthetic profiles are mostly limited to the statistical comparisons of synthetic and real data distributions on an aggregated level, there are often no concrete case studies or quantitative metrics that matches the intended use of the model. As suggested in [159], good performances of generative models on one statistical metric

doesn't imply the same level of success in other criteria so these models need to be evaluated concerning the intended end applications.

The following chapters address these research gaps through new analyses, data extrapolation, interpolation and synthesis models:

(1) Chapter 3 looks into the impacts of input data temporal resolution on the optimised costs and savings of a PV-battery optimisation model. The analysis also recommends the desirable granularity which achieves a good balance between computational costs and accuracies.

(2) Chapter 4 conducts the first clustering analysis taking account of both generation and consumption data of households, applies the clustering results to extrapolate the limited input data and uses the extrapolated data for battery sizing of residential PV households.

(3) A GAN based model is proposed in Chapter 5, which can simultaneously generate 5-minute residential gross/net meter PV and load power data, taking account of the correlations between on-site PV and load power. A practical end-use case study is presented using a residential battery sizing tool. The tool uses synthetic PV and load power data as inputs produced by DCGAN and is validated by a large evaluation set of 292 PV households.

(4) Chapter 6 presents the first SRGAN based model to produce 5-minute average PV generation and load power data from 30-minute/hourly average PV and load power measurements. The synthetically interpolated high temporal precision power data is validated in a PV integrated battery optimisation model, which for the first time, addresses the issue of applying coarse PV and load data in modelling residential PV battery systems.

Chapter 3

Evaluating the Impacts of Temporal Resolution and System Efficiency on PV-battery System Optimisation

3.1 Introduction

Optimising the charging/discharging activities of batteries is crucial to realise the full potential benefit of a PV-battery system. Many studies have sought to solve this sequential stochastic optimisation problem using various optimisation techniques such as linear programming, quadratic programming, dynamic programming and model predictive control.

As indicated in Table 2.3, which summarises applied temporal resolutions for the optimisation models used in the reviewed PV battery power scheduling literature, most studies use low-resolution forecasted/measured PV and consumption data. The use of low temporal resolution in an optimisation's objective function may lead to errors in estimated costs as realised costs are derived instantaneously in practice.

The literature remains unclear how low temporal precision could impact the optimised costs of objective functions in a PV battery power scheduling model. Therefore, it is worth looking more closely at the temporal resolution to understand how the optimised costs are affected. Figure 3.1 shows the PV & load power profiles for a single day with various temporal resolutions explored in this chapter. The goal is to find the most suitable resolution, resulting in a desirable balance between accuracy and computational costs.

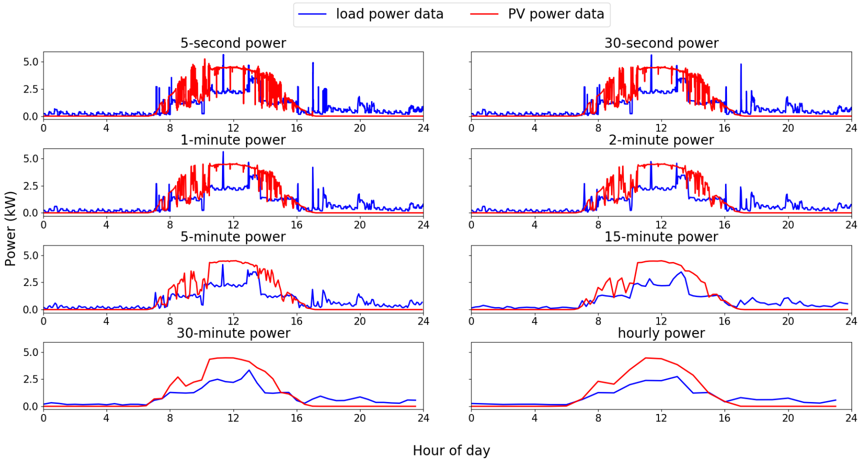

Figure 3.1: PV & load power profiles for a single day with various temporal resolutions explored in this chapter.

According to Table 2.2, several studies assume perfect battery conversions (i.e. the efficiency is assumed to be 100%). The majority of approaches incorporate a constant charging/discharging efficiency. Several studies have adopted a quadratic battery efficiency curve where the battery charging/discharging efficiency is dependent on the input/output power.

There is no published research looking at evaluating the impacts of various battery efficiency settings in a PV battery optimisation model to the best of the author's knowledge. Furthermore, there is a need to investigate the data from real PV battery systems to assess whether a linear battery efficiency model is sufficient.

3.2 Datasets

Three primary datasets collected by Solar Analytics, an Australian solar monitoring company [160] using Wattwatchers monitoring hardware [161], are used in this chapter:

(1) One year of 5-second PV and consumption data collected between August 2016 and August 2017, from 45 Australian residential customers.

(2) Up to one year of 30-second PV, consumption and battery energy data collected from 36 Australian residential battery customers who all have the same battery size and configuration.

(3) Up to one year of 30-minute battery application programming interface (API) data collected from the 36 residential battery customers mentioned above. The API data is directly provided by the battery manufacturer

and it includes the maximum usable capacity, 30-minute SOC, 30-minute battery charge and discharge energy.

3.3 Optimisation Algorithms

Two battery power scheduling optimisation models (rule-based (RB) and linear programming (LP)) are adopted to evaluate the impacts of various input data granularities. On the other hand, only the RB model is applied to evaluate the effects of different battery efficiency settings as the real battery systems included in this study are controlled by this algorithm so adopting this model allows us to make an empirical sensitivity analysis on battery efficiency by comparing real and estimated optimised costs.

The reasons for using RB and LP are twofold: 1. RB is a simple method which can be a baseline and LP could be the theoretical upper limit regarding the optimisation results. Under linear constraints and perfect forecasts, LPs optimisation results are already optimal. While other more complex models could better deal with non-linear constraints and non-perfect forecasts, they would still end up with the same results obtained by LP under this chapters assumptions. 2. The RB and LP models are fairly standard and straightforward in terms of implementation, this means the sensitivity analysis of this chapter could also be valid for other studies using LP and RB. However, for other more complicated models (e.g. DP), most studies have variances in terms of model assumptions and actual implementations even though they use similar approaches. This means it is hard to adopt a benchmark model in this analysis and to ensure the same findings would also apply for other studies.

3.3.1 Nomenclature

The nomenclature for the optimisation algorithms is shown in Table 3.1:

Table 3.1: Nomenclature for the optimisation algorithms.

Symbol	Definition
pv_t	Gross PV energy during interval t
$load_t$	Gross load energy during interval t
P_{max}	Rated maximum charging and discharging power (kW)
C_{total}	Total battery size (kWh)
SOC_{min}	Minimum value for state of charge
p_t^{import}	Import Tariff during interval t (AUD/kWh)
p_t^{export}	Export Tariff during interval t (AUD/kWh)
SOC_{start}	State-of-charge when we start our simulation
η_{ch}	Charging efficiency
η_d	Discharging efficiency
t	Time interval
m	Number of intervals in one year

h	Number of intervals in one full day
soc_t^{usable}	Usable capacity during interval t
net_energy_t	Net energy during interval t
b_t^{ch}	Energy transferred to battery during interval t (kWh)
g_t^{export}	Grid export during interval t (kWh)
$cost_t^{pv}$	Electricity Cost during interval t (AUD) without a battery
$cost_t^{batt}$	Electricity Cost during interval t (AUD) with an installed battery
soc_t	State of Charge at start of interval t (kWh)
b_t^{d}	Energy transferred from battery during interval t (kWh)
g_t^{import}	Grid import during interval t (kWh)
$degrad_c$	Battery degradation rate in total capacity (kWh/interval)
$degrad_p$	Battery degradation rate in maximum charging/discharging power (kW/interval) data

3.3.2 Rule-based (RB) Model

The rule-based model used in this work is a simple control algorithm that aims to maximise PV self-consumption. It has been considered and implemented for some studies (e.g. in [50]) and is used in practice at many installed batteries due to its simplicity and ease of implementation. Another reason to include this model is that all the real battery systems included in this study are controlled by this algorithm so adopting this model allows us to make an empirical sensitivity analysis on battery efficiency by comparing real and estimated optimised costs. A pseudo code of this algorithm is presented in Algorithm 1.

As shown in Algorithm 1, a linear degradation rate is assumed for both the maximum charging and discharging power and battery capacity, they are updated for each timestamp (t). Then there are two scenarios: 1. when PV generation exceeds load consumption, the battery is charged until it is full. At the same time, we also ensure the charge power is less than the battery limit; 2. when load consumption is larger than generation, the battery is discharged until depleted. Similarly, the algorithms ensure the discharge power is lower than the battery limit.

3.3.3 Linear Programming Model

Several researchers in this area have applied linear Programming (LP) as it can converge at a low computational cost and guarantee the solution is optimal if the optimisation problem is linear [25, 34, 55, 61, 68, 74, 78]. As we are running simulations at a high temporal resolution, LP is favoured to minimise computational costs. Table 3.2 demonstrates the mathematical cost function and convex constraints used in our LP formulation.

An optimisation planning horizon of 24 hours is included assuming perfect foresight of PV and consumption. It should be noted that in real-time, perfect forecasts are not possible. The reason for using a perfect forecast is that we

Algorithm 1 Pseudo Code for the rule-based model

```
Input $pv_t$, $load_t$, $P_{max}$, $C_{total}$, $SOC_{min}$, $p_t^{import}$, $p_t^{export}$, $SOC_{start}$ ▷ Input parameters
Input $\eta_{ch}$, $\eta_d$ ▷ Import charging/discharging efficiencies
$soc_o \leftarrow SOC_{start}$ ▷ Set initial SOC to $SOC_{start}$
for $t$ in (1, 2, ..., $m$) do ▷ for loop starts
    $soc_t^{usable} \leftarrow C_{total} \times (1 - SOC_{min}) - t \times degrad_c$
    $P_t^c \leftarrow P_{max} - t \times degrad_p$
    $net\_energy_t \leftarrow pv_t - load_t$ ▷ determine net load from gross PV and load energy
    if $net\_energy_t > 0$ then ▷ when there is excess solar, charge battery until full
        $b_t^{ch} \leftarrow min(net\_energy_t, P_t^c, (soc_t^{usable} - soc_{t-1})/\eta_{ch})$
        $g_t^{export} \leftarrow net\_energy_t - b_t^{ch}$
        $cost_t^{pv} \leftarrow -net\_energy_t \times p_t^{export}$
        $cost_t^{batt} \leftarrow -g_t^{export} \times p_t^{export}$
        $soc_t \leftarrow soc_{t-1} + b_t^{ch} \times \eta_{ch}$
    else ▷ when there is excess demand, discharge battery until depleted
        $b_t^d \leftarrow min(-net\_energy_t, P_t^c \times \eta_d, soc_{t-1} \times \eta_d)$
        $g_t^{import} \leftarrow -net\_energy_t - b_t^d$
        $cost_t^{pv} \leftarrow -net\_energy_t \times p_t^{import}$
        $cost_t^{batt} \leftarrow g_t^{import} \times p_t^{import}$
        $soc_t \leftarrow soc_{t-1} - b_t^d/\eta_d$
Output $\sum_{t=1}^{m} cost_t^{pv} \sum_{t=1}^{m} cost_t^{batt}$
```

Table 3.2: Formulation of the LP model

Objective Function	Minimise $J = \sum_{t=1}^{h}(g_t^{import} \times p_t^{import} - g_t^{export} \times p_t^{export})$
Variables	$b_t^{ch}, b_t^{d}, g_t^{export}, g_t^{import}$
Equality constraints	$pv_t + b_t^{d} + g_t^{import} = load_t + b_t^{ch} + g_t^{export}$
	$soc_t = soc_{t-1} + b_t^{ch} \times \eta_{ch} - b_t^{d}/\eta_d$
Inequality constraints	$b_t^{ch} \geq 0; b_t^{d} \geq 0; g_t^{export} \geq 0; g_t^{import} \geq 0$
	$b_t^{ch} \leq P_{max}; b_t^{d} \leq P_{max}$
	$0 \leq soc_t \leq soc_t^{usable}$

dont have benchmark real-time forecasting errors to incorporate into our optimisation model as the errors do vary a lot, depending on the adopted datasets and forecasting algorithms. Hence, we set the research question to be: what are the errors in optimised costs and savings of using coarse data when you have a baseline control algorithm (RB) and an optimal control algorithm (LP + perfect forecasts). These two algorithms set the upper and lower limits of the optimised electricity costs and battery savings, therefore they are best suited for the purpose of this analysis despite the fact that the upper limit may never be reached in real time.

Theoretically, the optimisation horizon of a PV battery system control problem is the lifetime of the system however, this is not adopted in this study for two reasons: (1) Longer horizons will exponentially increase the computational cost for optimisation algorithms such as LP. (2) For optimisation models that require forecasts of PV and load, it is not feasible to get forecasted data with adequate accuracy for a horizon equal to the lifetime of a system.

The Gurobi Optimiser [162] is used in Python to solve the 24-hour planning horizon, and then the derived control signals are implemented in the next day. Due to the high computational demands on solving 5 second and 30 second, we only perform our analysis on data with 1-minute temporal resolution and coarser. Moreover, we only consider the time-of-use (ToU) tariff structure for the LP model because, under a flat tariff structure, it is not viable to charge from the grid at a lower rate or perform other types of price arbitrage. Hence, maximising self-consumption like what we do in the RB model, is already the optimal control scheme.

3.4 Sensitivity Analyses

Two different sensitivity analyses are performed in this section: one is to evaluate the influences of various temporal resolutions on the optimised electricity costs and battery saving potentials. The other is to access the impacts of apply different battery efficiency settings on the battery optimisation results.

Table 3.3: Battery parameters used in the granularity and battery efficiency analyses

Parameter	Value used in granularity analysis	Value used in efficiency analysis
C_{total}	Unique optimal size	8.4 kWh
P_{max}	$0.4 \times C_{total}$	2.0 kW
SOC_{min}	20%	Derived from API data
$degrad_c$	0	Derived from API data
$degrad_p$	0	Derived from API data
SOC_{start}	50%	Derived from API data
η_{ch}	90%	Derived from API and energy flow data
η_d	90%	Derived from API and energy flow data

3.4.1 Battery Parameters

Table 3.3 illustrates the battery-related parameters used in the temporal resolution analysis and the battery system efficiency analysis. As the data granularity analysis uses the dataset consists of residential PV customers with no batteries, a proper battery sizing (more details are described in Section 3.4.2) is conducted for each household. In contrast, the battery efficiency analysis is performed on residential PV-battery sites; hence their actual battery parameters are adopted.

3.4.2 Analysis of Temporal Resolution

The main steps of the data granularity analysis are shown in Figure 3.2. 5-second residential PV and consumption data is first resampled into a few other lower resolutions (30-second, 1-minute, 2-minute, 5-minute, 15-minute, 30-minute and 60-minute) and then are fed into two different battery power scheduling optimisation models: a rule-based (RB) and a linear programming model. Then the optimisation models output two values: the yearly electricity cost without installing battery and the situation with installed battery. The yearly savings of operating the battery are found by subtracting the derived two yearly costs. We then use Eqn. 3.1 and Eqn. 3.2 to determine the relative errors to our finest resolution (i.e. 5-second) by comparing costs and savings of 5-second data with other coarser temporal resolutions.

$$Relative\ error\ in\ optimised\ costs = \frac{cost(lower\ resolution) - cost(highest\ resolution)}{cost(highest\ resolution)} \tag{3.1}$$

$$Relative\ error\ in\ savings = \frac{savings(lower\ resolution) - savings(highest\ resolution)}{savings(highest\ resolution)} \tag{3.2}$$

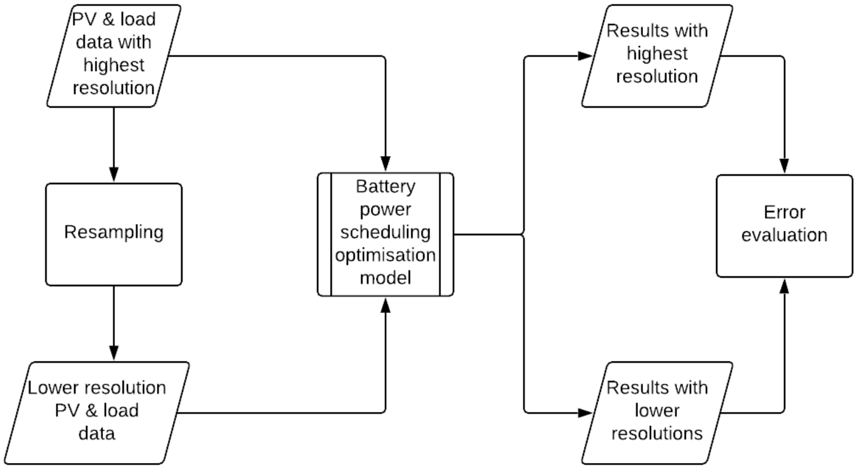
PV & load data with highest resolution
Resampling
Lower resolution PV & load data
Battery power scheduling optimisation model
Results with highest resolution
Results with lower resolutions
Error evaluation

Figure 3.2: The flowchart of the temporal resolution analysis.

Battery Sizing

For the temporal resolution analysis, we determine an optimal battery size for each residential PV customer without a battery by feeding their 5-minute PV and consumption data into a battery sizing model proposed in [163].

3.4.3 Sensitivity Analysis on Battery System Efficiency

The main process flow of the battery efficiency analysis is shown in Figure 3.3. PV and load data with various temporal resolutions and battery efficiency settings (single/dual/SOC tracking model) are fed into the same battery simulation model used in 3.4.2. Then the ground truth results determined using real-time battery system data are compared against the simulated results to evaluate the errors related to different battery efficiency settings.

Single Efficiency and Dual Efficiency

The first step of the battery efficiency analysis is to determine errors in estimated optimised costs and savings using a constant efficiency. Single efficiency is referred to as the situation when we assume charging efficiency equal to discharging efficiency (i.e. $\eta_{ch} = \eta_d$) and dual efficiency is when $\eta_{ch} \neq \eta_d$. Both efficiency settings have been previously applied in the literature list summarised in Table 2.2. In this study, we examine both scenarios separately by following these steps:

I Apply a linear curve fit on the 30-minute battery energy flows and capacity changes and then derive a single charging/discharging efficiency (see Eqn.

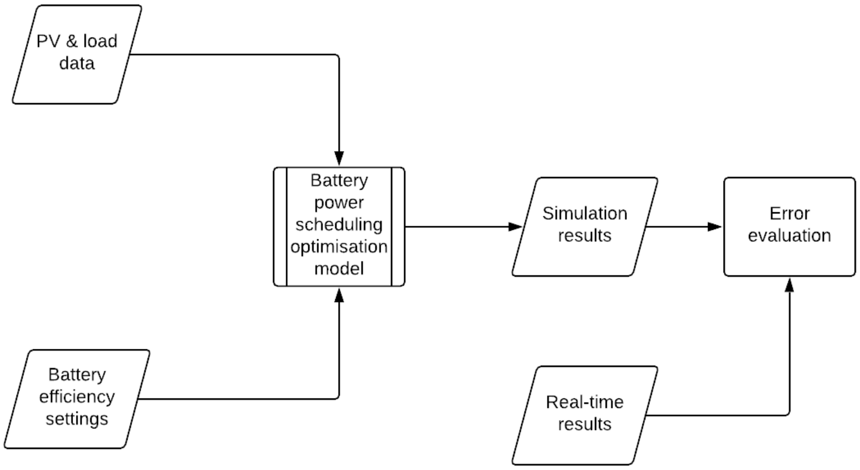
PV & load data
Battery power scheduling optimisation model
Simulation results
Error evaluation
Battery efficiency settings
Real-time results

Figure 3.3: The flowchart of the battery efficiency analysis.

3.3) and a dual charging & discharging efficiency setting (see Eqn. 3.4) for an individual battery customer.

For single efficiency setting:

$$\Delta C_{30min} = energy_{in} \times \eta_{single} - energy_{out}/\eta_{single} \tag{3.3}$$

For dual efficiency setting:

$$\Delta C_{30min} = energy_{in} \times \eta_{ch} - energy_{out}/\eta_{d} \tag{3.4}$$

II From the battery API data, Derive a linear capacity degradation rate ($degrad_c$) and a charging/discharging power degradation rate ($degrad_p$) for each battery site by fitting a linear curve on the time since a battery is installed and changes in the rated maximum usable capacity (C_{total}) and charging/discharging power (P_{max}).

III Determine the ground truth electricity costs and savings in various resolutions by applying battery, PV and load data of 36 residential battery customers.

IV Feed PV and consumption data from 36 battery customers and the parameters into the RB model to determine the estimated costs and savings.

Linear Regression SOC Tracking Model

A linear regression model formulated in Eqn. 3.5 is proposed to evaluate whether we could train a linear SOC tracking model using a limited amount of SOC and battery energy data instead of using data from a whole year like what is done in Section 3.4.3. Another initiative for this approach is that we suspect other features such as temperature and previous SOCs could enhance our results. So instead of just doing a linear curve fit for all the data we have for one site, we add features including previous SOCs, 30-minute ambient temperature and battery energy flows for 90 days and then implement the trained linear regression model in our RB simulation model for the rest of the data period. Therefore, instead of updating our SOC with constant efficiencies, we use the trained linear regression model. Finally, estimated optimised SOCs and optimised costs are compared against actual costs and battery API SOCs to see if we could obtain a satisfactory accuracy in estimated SOCs, electricity costs and savings.

$$soc_t = \boldsymbol{a} + \boldsymbol{b} \begin{bmatrix} soc_{t-1} \\ T_t \\ b_t \\ hour \end{bmatrix} \tag{3.5}$$

Where $\boldsymbol{a}$ and $\boldsymbol{b}$ are respectively the intercept and slope for the linear regression model. T_t is 30-minute ambient temperature in Celsius, b_t is the 30-minute battery AC energy flow in kWh and *hour* is the hour number derived from the 30-minute timestamp.

3.5 Tariff Structure

A flat tariff and a ToU have been considered for both temporal resolution and battery efficiency analyses. The adopted tariff rates are shown in Table 3.4.

Table 3.4: Tariff Rates for Flat and ToU

Flat Tariff ($AUD/kWh)	ToU Tariff ($AUD/kWh)		
	Peak (3pm to 9pm on weekdays)	Off-peak (10 pm to 7 am on weekdays & weekends)	Shoulder (all other times)
0.30	0.45	0.15	0.25

3.6 Results and Discussion

3.6.1 Impacts of Temporal Resolution on Optimised Costs and Savings

Relative errors in optimised costs and savings, which are illustrated in Figure 3.4 and Figure 3.5 for various granularities, clearly show underestimations in both optimised costs and savings derived from lower resolutions for RB and LP models. At an hourly resolution, compared to results with 5 second time

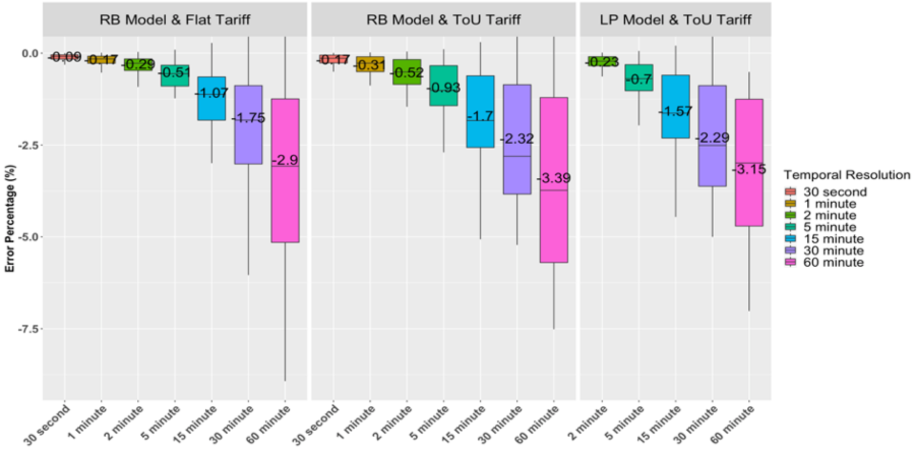

Figure 3.4: Percentage relative errors in yearly optimised costs for RB model with flat (left), ToU (middle) and LP Model with ToU (right) (numbers inside boxplots are the mean errors after excluding outliers).

interval, approximately 3% mean relative error is found in optimised costs across all included sites for the three explored scenarios. The RB model with ToU tariff seems to be slightly more sensitive to temporal precision compared to the flat tariff scenario. However, overall, the relative errors caused by coarser resolutions are consistent across both investigated optimisation models. On the other hand, the influence of granularity is much higher on the yearly electricity bill savings. As indicated in Figure 3.5, the mean relative errors in savings for 30-minute and 60-minute temporal resolutions could be as high as 9.11% and 12.6% for the RB model with flat tariff. The savings computed from the LP model are less sensitive to data granularity compared to the results from our RB model.

The results demonstrated in Figure 3.4 and Figure 3.5 give confidence in applying PV and consumption data of 5 minute or other finer temporal resolutions in PV battery scheduling optimisations. For our included residential sites, 5-minute data results in less than 1% and less than 4% underestimations in optimised costs and savings, respectively. Given that 5-minute data will not exceed the bandwidth limits of most smart meters in the current market, we recommend that 5-minute sampling rate is a viable option for PV battery power scheduling optimisation models.

3.6.2 Impacts of Constant Efficiency Settings on Optimised Costs and Savings

Table 3.5 illustrates the errors relative to the true cost calculated from 30 second real battery site import and export data. Although the real-time costs are derived instantaneously instead of every 30 second, from the results shown above in Figure 3.4 and Figure 3.5, we believe the resulting costs and savings

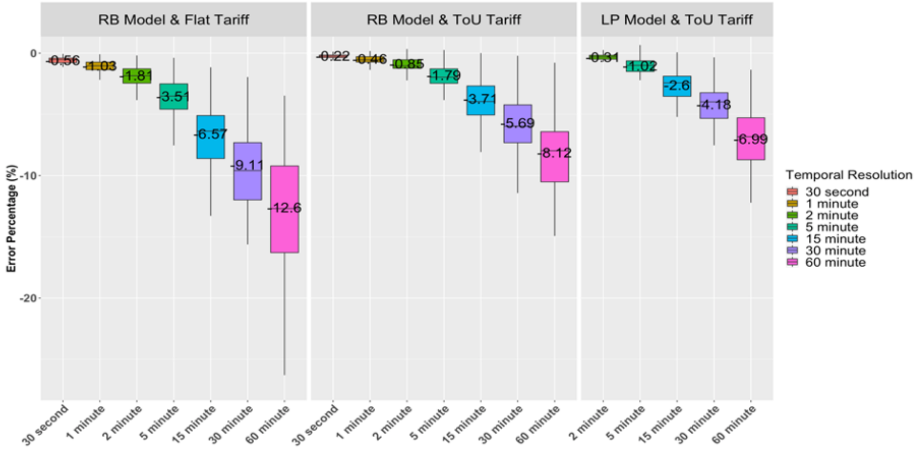

Figure 3.5: Percentage relative errors in yearly Savings for RB model with flat (left), ToU (middle) and LP Model with ToU (right) (numbers inside boxplots are the mean errors after excluding outliers).

from 30 second data can still be relatively close approximations to the real-time costs. Underestimations and overestimations can be observed, respectively in estimated costs and savings computed from our RB simulation model for both single and double efficiency settings. A few pronounced points are summarised below:

1. Applying constant efficiency settings results in significant errors in estimated costs and savings across all temporal resolutions.

2. Underestimations in optimised costs are larger with coarser input data which is consistent with what we found in Figure 3.4 and Figure 3.5, however the overestimations in savings are interestingly lower when we apply data with longer time intervals. We think this trend is caused by underestimations due to lower temporal resolutions cancelling out the overestimations effects of using constant efficiencies.

3. To examine our hypothesis on the cancelling effects, we recompute the relative errors shown in Table 3.6 by comparing simulated costs and savings with the true results generated from real imports and exports aggregated to each tested temporal resolution. So instead of comparing all the results to 30 second true costs and savings, we generate 1, 2, 5, 15, 30 and 60 minute true costs and savings from real imports and exports at these temporal resolutions to allow comparisons within the same temporal resolution so that we could minimise the impacts of temporal resolution in our efficiency analysis. As demonstrated in Table 3.6, we are now observing higher relative errors in savings and lower errors in optimised costs for coarser temporal resolutions.

4. It can be observed that the two efficiency settings (single and dual) make small differences in terms of errors in optimised costs and savings.

5. The included ToU tariff produces larger underestimations in optimised costs and smaller overestimations in savings than the results with flat tariff.

Table 3.5: Mean percentage errors relative to true yearly costs and savings

Tariff Structure	*Flat*		*ToU*	
Efficiency Settings	Single Efficiency	Dual Efficiency	Single Efficiency	Dual Efficiency
Mean percentage relative errors in optimised costs for various temporal resolutions (%)				
30 second	-8.01	-8.88	-9.51	-8.47
1 minute	-8.24	-8.28	-10.04	-8.92
2 minute	-8.58	-8.58	-9.84	-9.54
5 minute	-9.23	-9.16	-11.05	-10.69
15 minute	-10.35	-10.2	-13.1	-13.79
30 minute	-11.29	-11.11	-14.98	-15.66
60 minute	-12.79	-12.56	-17.45	-18.11
Mean percentage relative errors in savings for various temporal resolutions (%)				
30 second	19.06	20.46	14.27	15.31
1 minute	18.9	20.32	14.38	15.42
2 minute	18.64	20.08	14.5	15.57
5 minute	17.97	19.48	14.55	15.68
15 minute	16.48	18.03	14.19	13.92
30 minute	15	16.52	13.06	13.42
60 minute	12.64	14.12	10.88	11.86

Table 3.6: Mean percentage errors relative to true yearly optimised costs and savings with corresponding resolutions

Tariff Structure	*Flat*		*ToU*	
Efficiency Settings	Single Efficiency	Dual Efficiency	Single Efficiency	Dual Efficiency
Mean percentage relative errors in optimised costs for various temporal resolutions (%)				
30 second	-8.01	-8.88	-9.51	-8.47
1 minute	-8.05	-8.1	-9.76	-8.67
2 minute	-8.21	-8.24	-9.35	-9.08

5 minute	-8.64	-8.63	-10.31	-9.99
15 minute	-9.49	-9.41	-12.05	-12.74
30 minute	-10.2	-10.09	-13.66	-14.35
60 minute	-11.08	-10.92	-15.49	-16.15
Mean percentage relative errors in savings for various temporal resolutions (%)				
30 second	19.06	20.46	14.27	15.31
1 minute	19.06	20.5	14.5	15.57
2 minute	19.46	20.92	15.16	16.24
5 minute	20.7	22.16	16.79	17.85
15 minute	23.36	25.94	19.82	20.87
30 minute	25.9	28.53	22.63	23.7
60 minute	29.57	32.25	26.32	28.4

3.6.3 Evaluation of Linear Regression SOC Tracking Model

As illustrated in Table 3.7, a few error metrics have been implemented to evaluate the accuracy of our proposed SOC tracking model. Based on the mean absolute error (MAE) and median absolute error (MDAE), overall the linear regression has a relatively satisfactory accuracy on tracking SOCs. On the other hand, the mean square error (MSE), root mean square error (RMSE) and r-squared value suggest the model makes a noticeable amount of predictions that are quite far from the SOC labels collected from API. Errors in optimised costs and savings are mostly comparable to what can be observed in Table 3.6 however large overestimations which average at 44.24% are found in estimated yearly savings with flat tariff so there are not any noticeable improvements of including more input features.

Overall, there is still room for improvements in SOC tracking. It also appears that our model is performing exceptionally well at low SOC values but fails to make accurate estimations of high SOCs. As a result, significant overestimations are found in estimated savings.

Table 3.7: Error metrics for estimations of SOCs and errors in optimised costs and savings

Error metrics for estimations of SOCs	Mean Value	Errors in optimised costs and savings	Mean Error Percentage
Mean absolute error	4.79	Error in yearly optimised costs with flat tariff	-14
Root mean square error	12.34	Error in percentage for yearly savings with flat tariff	44.24
Median absolute error	1.25	Error in percentage for yearly optimised costs with ToU tariff	-16.98

R squared value	0.82	Error in percentage for yearly savings with ToU tariff	29.75
Mean square error	126.31		

3.7 Summary

In this chapter, a sensitivity analysis is performed on the influences of applying coarser PV/consumption data and constant battery efficiencies in a PV battery power scheduling optimisation model. It is shown in Figure 3.4 and Figure 3.5 that low temporal resolutions can lead to noticeable underestimations in both optimised costs and savings for all optimisation scenarios explored in our approach. Then it can be concluded that 5-minute temporal resolution is sufficient to compute results with a good level of accuracy. Furthermore, as illustrated in Table 3.6, the sensitivity investigation on applying constant battery efficiencies demonstrates significant underestimations in estimated electricity costs and even larger overestimations in electricity bill savings. It should also be noted that a cancelling effect is found when implementing both coarser data and constant efficiencies, the resulting errors in savings are reduced, as shown in Table 3.5. Furthermore, Table 3.7 indicate that the linear regression model that includes more features such as temperature and previous SOCs did not make any noticeable improvements in reducing relative errors of optimised costs and savings.

Chapter 4

Battery Sizing Model using Net Meter Energy Data Clustering

4.1 Introduction

In recent years, driven by the technology cost reductions and government incentives, the industry has witnessed rapid rollouts of rooftop PV systems in the residential sector. Australia leads the world in residential PV penetration, as of the end of 2015 with 15.22% of households owning a rooftop solar system [164] and this number has increased to 21.1% at the end of 2017 [165]. Several European countries also have considerable amounts of residential solar penetration, such as Belgium (7.45%), Germany (3.72%) and the UK (2.52%) [164].

Although the generous feed-in tariffs have accelerated the adoption of residential PV, most have been cancelled or reduced in various countries and regions due to the reduction in technology costs [166]. Since the solar feed-in tariffs are now lower than the general import tariffs in many regions, the net metering scheme is considered a viable option to reduce the electricity costs for PV consumers. Net metering also brings opportunities to the energy storage market as batteries can now provide more benefits such as peak shaving, increasing PV self-consumption and price arbitrage. Before going ahead with purchasing a battery, the financial returns or other metrics regarding the battery capabilities need to be carefully evaluated.

Although many techno-economic simulation models have been proposed, the practicability of these approaches remains questionable due to two main reasons:

(1) Many studies use synthetic household PV or load data, resulting in misleading simulation results [118]. Individual households could have various consumption profiles and solar systems with different orientations, tilts or shading conditions. Moreover, it is essential to use both generation

and consumption data of actual solar customers as their consumption behaviours may change after the PV installations [137]

(2) A minimum amount and high-quality of input PV / weather and load data is often required to build a robust model, whereas such data might not be available in practice.

One approach to potentially address the above issues is to perform data extrapolation using a customer's consumption and generation patterns extracted from the limited historical data.

Generally, the knowledge of users' electricity consumption patterns is applied to develop tariff structures [124], demand response strategies [125], load forecasting and planning models [128, 129]. In reality, in order to gain a good understanding of the consumption and generation profiles for solar customers, it is vital to conduct the clustering analysis on both the generation and consumption data. Motivated by these facts, this chapter introduces a battery sizing model for residential PV customers using net meter energy data clustering.

The main purpose of this chapter is to develop a model for solar customers with net meter arrangements and limited amounts of historical consumption and generation data to decide on the most optimal battery size.

4.2 Battery Sizing using Net Meter Clustering

The methodology of the net meter clustering approach (shown in Figure 4.1) can be separated into four parts:

1. The dataset is prepared for net meter clustering and separated into a training set and an evaluation set. The training set is used for fitting the parameters of the clustering and regression models mentioned below and the evaluation set is used to evaluate the performance of the proposed model.

2. K-means clustering is applied to cluster net meter energy data separately for various seasons; Summer, Autumn, Winter and Spring. For each household, the seasonal cluster distributions are determined, which specify each household's percentages of seasonal net meter profiles partitioned into each seasonal cluster.

3. Regression models are trained on the obtained seasonal cluster distributions and used to extrapolate the seasonal cluster distributions of the new input net meter energy data at a given length.

4. The extrapolated data, battery and economic parameters are fed into a battery simulation model that produces the optimal battery sizes for the customers in the evaluation set.

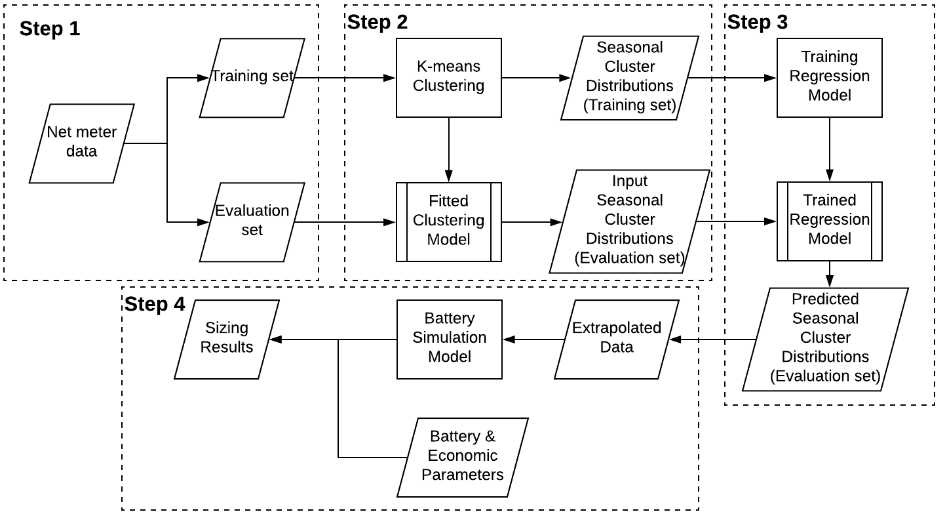
Step 1
Net meter data
Training set
Evaluation set
Step 2
K-means Clustering
Seasonal Cluster Distributions (Training set)
Fitted Clustering Model
Input Seasonal Cluster Distributions (Evaluation set)
Step 3
Training Regression Model
Trained Regression Model
Predicted Seasonal Cluster Distributions (Evaluation set)
Step 4
Sizing Results
Battery Simulation Model
Extrapolated Data
Battery & Economic Parameters

Figure 4.1: The flowchart of the proposed battery sizing model using net meter clustering.

For comparing the net meter energy clustering approach, an alternative naive prediction method (see Section 4.3) is also implemented. To evaluate the performances of the two methods, the battery sizing results are also derived for the ideal case where a whole year's measured data is provided to the battery simulation model instead of extrapolated data. This allows us to determine errors in the battery sizing results for the two implemented modelling approaches.

4.2.1 Step 1 - Pre-clustering Step

Data Collection

The data used in this chapter was also collected by Solar Analytics [160] using Wattwatchers monitoring hardware [161]. The dataset includes 5-minute gross PV and consumption data collected between December 2016 and December 2017 from 2779 Australian solar households. Using solar and load data collected from the same households, the dataset can take account of the impacts of domestic solar generation on consumption behaviours. These customers have adequate amounts of data: the overall amount of missing data is less than 3% and the customer with the most missing data has 7% of data missing. To deal with missing data, days with more than two hours of missing data are excluded from the dataset. The DC solar system ratings of these customers are also recorded, and these rooftop PV systems have been performing normally without any significant system faults within this period. To construct a net meter dataset from gross meter data, the gross PV and consumption data are converted to net meter energy data using Eqn. 4.1 and then the net meter energy data is resampled to 30-minute temporal resolution.

$$net_{energy} = pv_{energy} - load_{energy} \tag{4.1}$$

Before applying any clustering, load curve normalisation has been applied in some previous load clustering studies [127, 135, 167, 168]. On the other hand, some consumption clustering studies use raw consumption data [134, 137, 169]. In this study, after carrying various simulations, using normalised data did not produce as good results as the raw data so it was decided to present results for only the raw data.

Data Split

To properly evaluate our battery sizing model, the dataset is divided into a training set and an evaluation set. Clustering is only performed on the training set, which includes 2517 randomly-selected customers. The remaining 262 households included in the evaluation set were treated as new customers to evaluate the robustness of the proposed battery sizing model against limited input data.

4.2.2 Step 2 - Net Meter Clustering

As seasonality generally exists in both solar generation and consumption data, the dataset is divided into four seasons and clustering is performed on each of them. Four seasons are defined as follows for Australia [170]: Summer: December to February, Autumn: March to May, Winter: June to August, Spring: September to November. Each daily profile of the customers in the training set was used in clustering to capture most information during the clustering process.

Clustering Algorithm

The K-means algorithm [171] is used for the net meter profiles as it has been proven to be simple yet effective in previous load clustering studies [127, 168] and furthermore, it converges quickly, which is a great advantage for large datasets [172].

Clustering Evaluation

In the previous clustering studies [124, 134], clustering validity indicators (CVIs) have been used to evaluate the performance of consumption data segmentation and to find the optimal number of clusters. On the other hand, the end-use application of this study is choosing the optimal battery size and approximating potential savings. Therefore, the number of clusters was chosen according to the minimum errors obtained for these tasks. In the meantime, to see whether there is any relationship between the CVI and errors obtained in battery sizing results, Davies-Bouldin index (DBI) [173] is also computed.

Seasonal Cluster Distributions

After separating the training data into four seasons, daily net meter profiles are clustered into various seasonal clusters. As a result, the distributions of each household's clustered net meter profile are calculated for these seasonal clusters. They simply describe each household's percentages of seasonal net meter profiles assigned to each seasonal cluster. It should be noted a seasonal cluster distribution does not require the whole season of data to be computed. In fact, it can be calculated for any period within the season. For instance, when a new customer has 30 days of net meter energy data in Summer where 18 daily profiles are grouped into cluster 1 and 12 days are in cluster 3. The Summer cluster distribution of this household is 60% (18/30) in cluster 1, 40% (12/30) in cluster 3 and 0% for other seasonal clusters.

Seasonal cluster distributions reveal the typical seasonal net meter patterns and their occurrences for a household at a given period. Therefore, when two households have similar seasonal cluster distributions within a period, they show similar net meter profiles in the same period. Moreover, the seasonal cluster distributions can be used as extracted features to predict seasonal distributions of other unknown periods which will be shown in the following sections.

4.2.3 Step 3 - Seasonal Cluster Distribution Prediction

This step trains a machine learning model to predict the seasonal cluster distributions for new customers with limited net meter energy data. In this chapter, multivariate linear regression and random forest regression techniques were compared. Feature selection and hyperparameter tuning are adopted to enhance the performance of regression models. The main steps to train the machine learning model are shown in Figure 4.2. For both regression techniques, feature selection is applied using the regression model with default hyperparameters and then parameter tuning is performed to select hyperparameters that lead to superior regression results. After that, the tuned model and selected features are used for model training. Finally, after training, the trained model is used to predict seasonal cluster distributions.

For each predicted season/period, the model searches for a customer that shows the most similar seasonal cluster distributions and has full length of data. In particular, this is done by finding the customer in the training set with the shortest Euclidean distance in terms of seasonal cluster distributions in the predicted season/period. This customer's data is then used as the seasonal extrapolated data for the new customer.

In terms of the length of data from the new customers, three options are considered; a single month, a single season or two random seasons. Also to evaluate the impacts of applying different months/seasons as inputs, all the input data scenarios in Table 4.1 are tested. Finally, the model output values are the seasonal cluster distributions for the remaining period of a year.

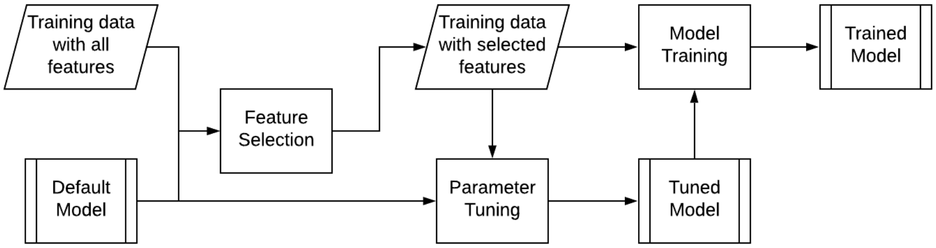

Figure 4.2: Main steps of the regression model training.

Table 4.1: Tested Input Data Scenarios

Input Data Length	Tested Input Data Scenarios
One month	Month number in [1 - 12]
One season	Seasons in [Summer, Autumn, Winter, Spring] ([1 - 4])
Two seasons	Two-season combinations in [1&2, 1&3, 1&4, 2&3, 2&4, 3&4]

Features

The input features used for predicting seasonal cluster distributions are listed in Table 4.2, which contains each household's: DC solar system size, state code, the daily averaged 30 minute net meter energy and the seasonal cluster distributions of the net meter energy data for the given period.

To compute the averaged daily net meter energy, the averaged energy is determined within the known data period for each 30-minute interval of a day. The dataset includes customers from 6 Australian states/territories: Australian Capital Territory, New South Wales, Queensland, South Australia, Victoria and Western Australia. They are converted to integers from 1 to 6.

Table 4.2: Features used for Regression

Feature Name	Symbol
seasonal cluster percentages of season i with n clusters	$p^1_{season_i}, p^2_{season_i}, ..., p^n_{season_i}$
mean 30 minute net meter energy (Wh)	$e_1, e_2, ..., e_{48}$
PV system size (kW)	pv_{size}
state code	s_{code}

If the input data has overlapping periods between different seasons, for example, if the given input data has 60 days which include 20 Winter days and 40 Spring days, the input seasonal cluster percentages would include both Winter and

Spring cluster percentages for the provided data. The predicted values would be the seasonal cluster distributions in Summer, Autumn and the remaining periods of Winter and Spring.

Multivariate Linear Regression

Multivariate linear regression (LR) [174] is a machine learning approach where multiple independent variables are used to predict multiple dependent variables. The regression problem in this study can be formulated by Eqn. 4.2 using this technique. The ordinary least square method, which minimises the squared differences between training labels and predicted values, is applied to estimate the parameters in Eqn. 4.2.

$$Y_{i,n} = \beta_{0,n} + \beta_{1,n}X_{i,1} + \beta_{2,n}X_{i,2} + \beta_{3,n}X_{i,3} + \ldots + \beta_{p,n}X_{i,p} + \epsilon_{i,n} \tag{4.2}$$

Where $Y_{i,n}$ is the predicted proportion of days clustered into seasonal cluster n for a sample i, $X_{i,j}$ is the jth feature used for a sample i, $\beta_{j,n}$ is the jth parameter estimating $Y_{i,n}$ and $\epsilon_{i,n}$ is the error term. The Python implementation of this model is used in this thesis [175].

Random Forest Regression

Random Forest (RF) is an ensemble machine learning method which trains multiple decision trees on different random subsets of the training data [176]. The adopted RF model uses the bootstrap aggregating (bagging) technique for training, where random subsets are drawn with replacements. Each subset has the same sample size as the original training set [177]. Given the training data X_{train} and the output label Y_{train}, by using bagging, N random subsets are generated from X_{train} and Y_{train} (denoted as X_d and Y_d). For each sampled subset, a decision tree f_d is trained using X_d and Y_d. When predicting a new sample after training, the RF model will aggregate the predictions from these decision trees. For regression tasks, the aggregation function takes the mean of the predictions by various decision trees (shown below in Eqn. 4.3).

$$Y_{test} = \frac{1}{N}\sum_{d=1}^{N} f_d(X_{test}) \tag{4.3}$$

Where Y_{test} denotes predicted labels of the test set, X_{test} is the input test data. As a result, compared to a single decision tree trained with the whole dataset, RF generally performs better. It reduces the model variance whilst resulting in similar bias errors [178]. The Python implementation of this model [175] is used and N is set to 100 which means results from 100 decision trees are aggregated within the RF model.

Feature Selection

For the linear regression model, a Least Absolute Shrinkage and Selection Operator (Lasso) regression analysis [179] is applied, which performs both L1 regularisation and feature selection. It penalises the absolute sum of coefficients, as a result, the regression coefficients for some features shrink towards zero and hence they are filtered out from the model.

The Boruta algorithm [180] is applied to select features for the RF model as it previously outperformed other RF feature selection approaches [181]. The algorithm randomly performs permutation on all features and train the RF model using both the original and shuffled features [180]. For each original feature, a statistical test is conducted which computes the confidence towards a better importance value compared to the maximum importance value of the shuffled features. Features with significantly higher importances are marked as important features whereas features with smaller importances are removed. Then the process will re-iterate until all features are categorised as confirmed/rejected or until a certain number of iterations is reached. In this study, the Python implementation of Boruta [182] is applied and the maximum number of iterations is set to 30. It is also suggested in [182] that the original threshold where a real feature needs to have better importance than all the shuffled features can sometimes be too stringent so the percentile parameter is set to 80% which means true features will pass the statistical test when its importance is higher than 80% of the shuffled features.

Parameter Tuning

To achieve better performances from our regression models, the hyperparameters of the models are optimised by using random search along with 10-fold cross validation (CV). Compared to other hyperparameter optimisation approaches such as grid search and manual search, random search has proven to be more efficient in computational costs [183]. The hyperparameters tuned for the linear regression and RF models are shown in Table 4.3. Some of the default parameters are selected from their default values set by sklearn [175] and the others are selected by experience to create a loosely tuned default model for feature selection.

In a 10-fold CV, it randomly splits the training set into 10 equal sized subsets. Nine subsets are used as training data and the remaining subset is evaluated once as a test set. This validation process is repeated ten times, where each time a different subset is used as a test set, after that the averages and standard deviations of the mean squared error (MSE) in seasonal cluster proportions are computed. Then the hyperparameters that yield the lowest averaged MSEs are chosen.

4.2.4 Step 4 - Battery Sizing Model

After predicting the seasonal cluster distributions for all the listed input data scenarios in Table 4.1 and extrapolating the net meter energy data for the unknown period, the battery sizing results are determined by feeding the extrapolated net meter profiles for the entire year to a battery simulation model which is described below in detail.

Table 4.3: Tuned Parameters for Lasso model and RF model [175]

Parameter in sklearn	Parameter Description	Tuning Range	Default Value
Lasso Regression Model			
alpha	constant to multiply the L1 regularisation term	float in [0 - 1]	1.0
RF Model			
max_features	the number of randomly drawn input features when considering the best split	n, $\sqrt{n}$, or 1/3 n (n is the number of all features)	n
max_depth	The maximum depth of the decision trees	integer in [2 - 12]	6
min_samples_leaf	The minimum number of required samples at a leaf node	integer in [1 - 12]	2
min_samples_split	The minimum number of required samples to make an internal split	integer in [2 - 12]	2

Model Parameters

Key parameters used in the battery simulation model are listed in Table 4.4. The model simulates annual realistic battery operations and computes battery sizing results using the listed battery and economic parameters for the three approaches discussed above: our net meter clustering approach, the naive prediction method and the ideal case where the whole year's data is provided.

Rule-based (RB) Model

The battery charging/discharging is assumed to follow the same rule-based algorithm described in Algorithm 1 that has the main objective of maximizing solar self-consumption.

Determine optimal battery size

The optimal battery size is determined by searching for the value which maximises the Net Present Value (NPV) at the end of the battery lifetime, defined below in Eqn. 4.4. The current residential batteries in the market are between 1 to 15 kWh [184], so this range is used for the grid search (i.e. 16 values in total including 0 kWh which means no battery is installed). The averaged warranty provided by manufacturers is around 10 years [184] however, adopting a

Table 4.4: Battery Simulation Parameters

Parameter	Definition	Values
Battery Specifications		
C_{total}	Total Battery Size (kWh)	1-15 kWh
P_{max}	Rated maximum charging/discharging power (kW)	$0.4 \times C_{total}$
SOC_{min}	Minimum value for state of charge	20%
SOC_{start}	SOC when simulation starts	0%
η_{ch}	Charging efficiency	90%
η_{d}	Discharging efficiency	90%
Economic Parameters		
$n_{lifetime}$	battery lifetime	15 years
$rate_{discount}$	discount rate	0.03
$saving_{degr}$	yearly reduction in saving due to battery degradation	0.05
Tariffs (in AUD / kWh)		
p_{flat}	flat import tariff rate	\$ 0.30 / kWh
p_{peak}	peak import tariff rate	\$ 0.45 / kWh
$p_{shoulder}$	shoulder import tariff rate	\$ 0.25 / kWh
$p_{offpeak}$	off-peak import tariff rate	\$ 0.15 / kWh
p_{fit}	flat feed-in tariff rate	\$ 0.11 / kWh

10-year lifetime makes it infeasible to install batteries for most solar customers, even with a low battery price scheme. Therefore 15 years is adopted for the maximum lifetime of a battery in the simulation model so it would be easier to compare errors in optimal battery sizes for the two tested approaches.

$$\begin{aligned} NPV &= -cost_0 + \sum_{t=1}^{n_{lifetime}} \frac{saving_t \times (1 - saving_{degr})^t}{(1 + rate_{discount})^t} \\ &= -(c_{batt} \times size_{batt} + c_{install}) \\ &+ \sum_{t=1}^{n_{lifetime}} \frac{(pcost_t - bcost_t) \times (1 - saving_{degr})^t}{(1 + rate_{discount})^t} \end{aligned} \tag{4.4}$$

Where $cost_0$ is the total capital costs including costs of battery, inverter and installation. It is assumed the costs of a battery and a new multimode inverter increase by c_{batt} when adding 1 kWh of battery capacity and installation costs ($c_{install}$) remain the same. $saving_{degr}$ is a degrading factor on yearly battery savings, it is assumed that savings will reduce annually by 5% due to battery degradation to save our computational costs. This is an arbitrary estimated parameter determined by the general guaranteed end lifetime usable capacity which is $60\% \approx (1 - 5\%)^{10}$ [185]. Yearly saving ($saving_t$) is simply derived by subtracting the yearly cost ($pcost_t$) without installing a battery and annual costs after the battery installation ($bcost_t$).

4.3 Alternative Comparative Approach

For comparing our net meter clustering approach, an alternative method is adopted. In this method, instead of performing clustering on the new customers, a naive forecasting approach is applied which finds another customer from the training data with the most similar net meter profile in the known period, measured by finding the shortest Euclidean distance between net meter profiles. Then for predicting the remaining periods of the year for the new customer, the net meter energy data of the closest site is simply used as a naive prediction. Furthermore, to evaluate the performances of these two prediction models, the battery sizing results are also derived for the ideal case, where a whole year of real monitored net meter energy data is provided to the battery simulation model. This ideal case allows us to compute the errors in optimal battery sizes, net present values, yearly battery savings and electricity costs for the net meter clustering case and the naive forecasting approach. To properly assess these two approaches, the battery sizing results are only computed for the evaluation set. It has not been used for fitting the parameters of the clustering and regression models.

4.4 Clustering Results

For each season, the Davies-Bouldin Index (DBI) is calculated for adopting various numbers of clusters to cluster the training set of 2517 customers where a smaller value of DBI indicates a better clustering outcome. As shown in Figure 4.3, seasonal DBIs improve as the numbers of seasonal clusters increase. However, making too many clusters could result in clustering results that are not desirable for the post clustering applications; hence user inspection is often required. Authors in [9] suggested locating the "elbow points" in a DBI curve as the numbers of clusters in terms of segmentation quality since DBI improves little beyond these points. The same approach is adopted in this chapter, as a result, Figure 4.4 illustrates the seasonal cluster centroids using the optimal numbers of seasonal clusters determined by DBI.

For Summer clustered groups, cluster 1 and 5 have similar peaks of grid import and export whereas in cluster 4, evening load is much higher than the export around noon. Customers who have a majority of the net meter profiles in cluster 3, 8 and 10 have higher solar generation than night-time and early morning consumption. Electricity import and export are both at low levels in cluster 2. On the other hand, in cluster 6, 7 and 9, on average there is no export mainly due to higher levels of daytime consumption. Overall, net meter profiles in cluster 2, 3, 8, 10 are more likely to benefit from small-size batteries as they have low level imports, whereas larger batteries are more suitable for profiles in cluster 1, 4, 5 which have considerable amounts of imports and exports. For cluster 6, 7 and 9, energy storage is not a good option as on average there is no excess PV generation.

In Autumn, high export and low night-time grid import is observed in cluster 4, 5 and 9 where cluster 4 has a lower export compared to cluster 5 and 9. Cluster 2, 3, and 7 all show a considerable amount of import and export however cluster 3 has a higher night-time load compared to the other two. Centroids of cluster 6 and cluster 8 both have zero net export with morning and evening consumption peaks, whereas cluster 1 has small amounts of import and export. Small-size batteries seem to be beneficial for net meter profiles in cluster 1, 4, 5, 9 where a small amount of energy is required from the battery to cover the consumption in non-solar periods. Cluster 2, 3, and 7 will get more savings from larger battery sizes whereas it would be hard to utilise batteries for profiles in cluster 6 and 8 as there is no excess generated energy.

For Winter, a few groups (cluster 4, 8 and 9) have low night-time consumption and noticeable amounts of exports. On the other hand, three cluster centroids (5, 7, 11) have zero net generation. Low export and high import is observed in cluster 1, 3, 6 and 10, where cluster 1 and 3 show higher night-time consumption while cluster 6 and 10 have higher morning load. Relatively high export and import are shown in cluster 2. Overall, most net meter profiles can only utilise a small amount of battery capacity as they either have low net consumption (cluster 4, 8 and 9) or their generation is low (cluster 1, 3, 6 and 10). Net meter profiles in cluster 5, 7 and 11 have insufficient energy to charge batteries whereas cluster 2 can fully utilise a medium or large residential battery.

In Spring, centroids of cluster 1, 4 and 10 show zero export, however cluster 2, 3 and 9 have significant grid exports. Three clusters (5, 6 and 8) have considerable exports and imports whereas import and export levels are both low in cluster 7. Small batteries can be more beneficial for net profiles in cluster 2, 3, 7 and 9, whereas larger batteries can be fully utilised for cluster 5, 6 and 8. On the other hand, energy storage can not be utilised in cluster 1, 4 and 10.

4.5 Prediction Results

4.5.1 LR model vs RF model

Figure 4.5, Figure 4.6 and Figure 4.7 illustrate the mean squared errors (MSEs) in predicted seasonal cluster proportions for various evaluated input data lengths using five seasonal clusters for each season as an example. 10-fold cross validation is performed to generate MSEs for each randomly selected subset of the training set. This allows us to generate boxplots to display the distributions of MSEs. The results indicate the Random Forest (RF) model outperforms the Linear Regression (LR) model for all the evaluated scenarios therefore, this model will be adopted for the data extrapolation process.

When using monthly or seasonal data as input, Autumn tends to produce the best regression results and has much smaller MSEs compared to the scenarios

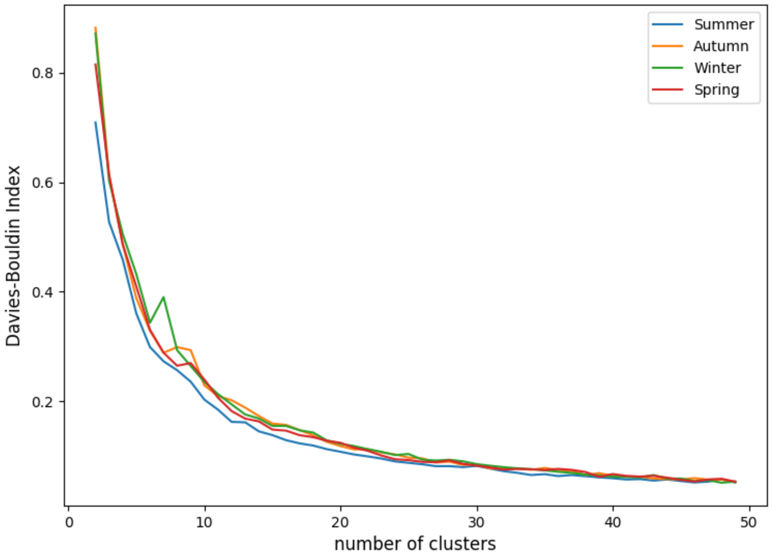

Figure 4.3: Davies-Bouldin Index for adopting various numbers of clusters each season using raw data.

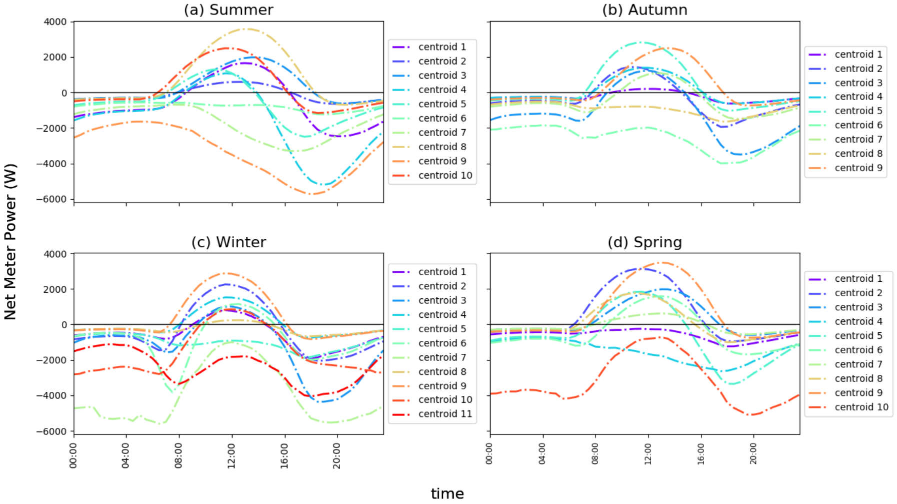

Figure 4.4: Seasonal unnormalised cluster centroids in (a) Summer, (b) Autumn, (c) Winter, and (d) Spring using optimal numbers of clusters.

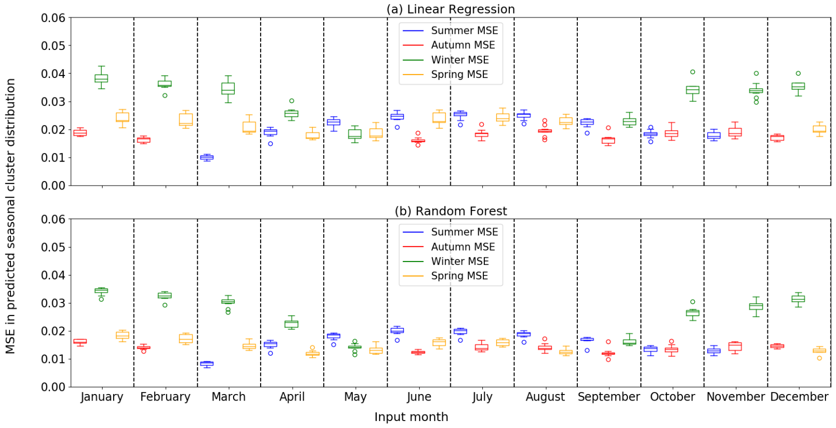

Figure 4.5: Mean Squared Error (MSE) in predicted seasonal cluster distributions using one month of input net meter energy data for the adopted (a) linear regression model, (b) random forest model.

using Winter or Summer. This is likely since Autumn has a more balanced generation and consumption, whereas Winter and Summer have either dominant generation or consumption.

For predicting new households net meter profiles by applying single monthly data as inputs; Winter seasonal cluster proportions seem to be the hardest seasonal cluster distributions to predict while it is much easier to determine these values in Autumn and Spring. This is likely caused by low irradiance in Winter which causes the Winter cluster distributions to be heavily influenced by household consumptions. In contrast, solar generation is more dominant within the input data period.

January seems to be the worst month for predicting other seasons as it generates the highest MSEs in predicted cluster proportions. It is interesting to note that to predict cluster distributions in Spring, April produces the best results whereas for other three seasons, the months adjacent to the predicted seasons have the lowest MSEs.

It is also interesting to note that in some cases when months adjacent to the predicted seasons are used (e.g. using May to predict cluster distributions in Winter), predicting with one month of data results in lower MSEs compared to one whole season of input data. The reason for that is probably months adjacent to the predicted season have quite similar consumption and generation patterns to the predicted season. Adding other months results in worse input features (i.e. the seasonal cluster distributions and mean net meter energy values).

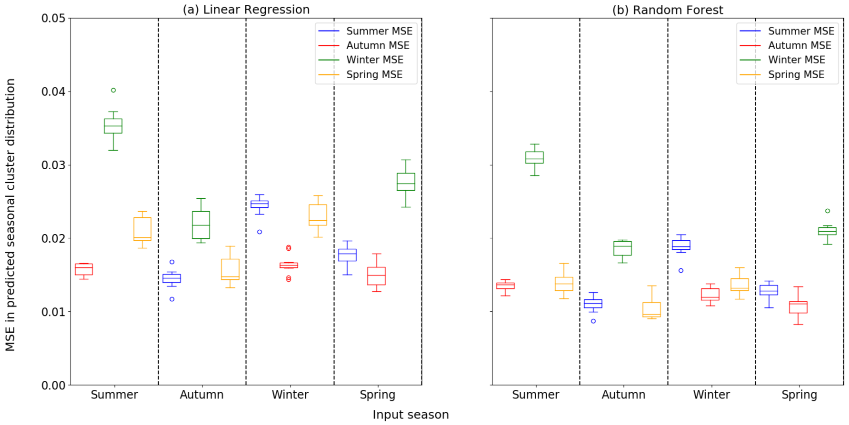

Figure 4.6: Mean Squared Error (MSE) in predicted seasonal cluster distributions using one season of input net meter energy data for the adopted (a) linear regression model, (b) random forest model.

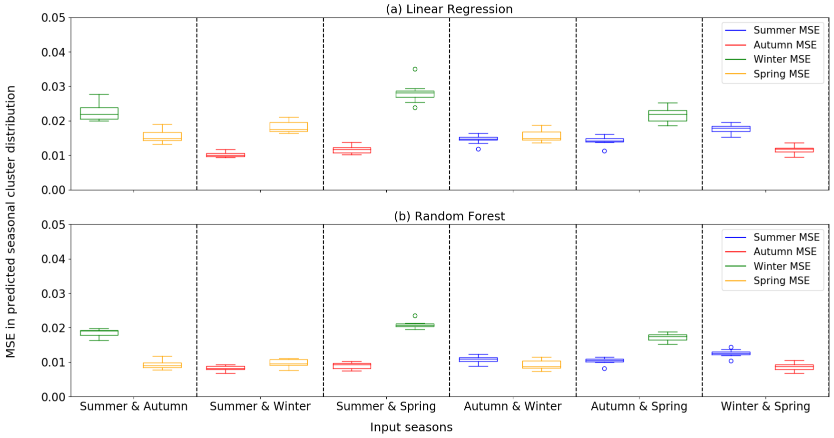

Figure 4.7: Mean Squared Error (MSE) in predicted seasonal cluster distributions using two seasons of input net meter energy data for the adopted (a) linear regression model, (b) random forest model.

4.5.2 MSE vs number of clusters

For each input data length, the MSEs are averaged for each tested scenario and plot them against various number of seasonal clusters. The same number of clusters are applied in each season to avoid creating too many combinations. As shown in Figure 4.8, the RF model still outperforms the LR model when using other numbers of seasonal clusters. MSEs in predicted seasonal cluster distributions are reduced when the number of clusters increases in each season. After the number of seasonal clusters reaches 30, the improvements in the averaged MSE slow down significantly.

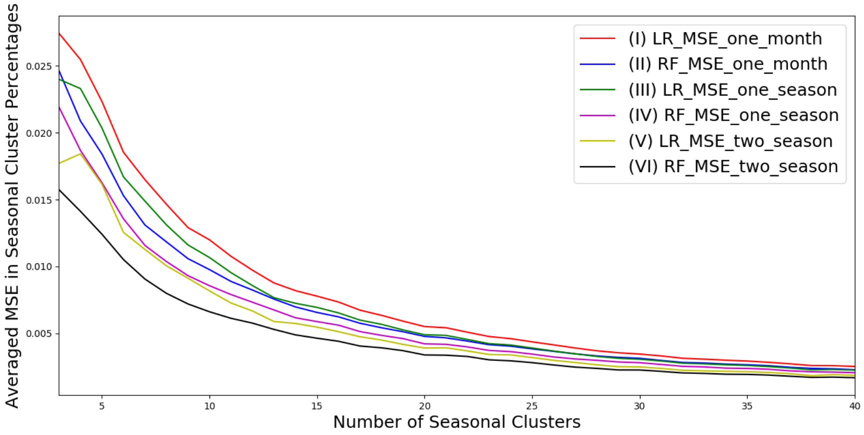

Figure 4.8: MSEs vs no seasonal clusters when applying (I) one month of input data and LR, (II) one month of input data and RF, (III) one season of input data and LR, (IV) one season of input data and RF, (V) two seasons of input data and LR, and (VI) two seasons of input data and RF.

4.5.3 Feature Selection and Parameter Tuning

Feature selection and parameter tuning both have improved the regression results. For example, for the specific case where data in Summer is inputted to predict seasonal cluster distributions in Spring and 5 clusters are used for each season. Forty-two features are selected after applying the Boruta algorithm on the default RF model, then parameter tuning is performed. As a result, compared to the original RF model with default features that produced 10-fold cross-validation MSE of 0.01412 (mean) ± 0.00154 (standard deviation), the MSE derived after feature selection and parameter tuning is 0.01389 (mean) ± 0.00139 (standard deviation).

4.6 Battery Sizing Results

4.6.1 Errors in Yearly Costs and Savings

Figure 4.9 and Figure 4.10 show the normalised root mean square errors (NRMSE) in yearly savings and costs against the number of seasonal clusters for both naive forecasting case and net meter clustering case with different input data lengths. The evaluated range for the number of seasonal clusters is from 3 to 40. The equivalent numbers of clusters are used for each season to avoid creating too many combinations for our analysis. Various battery sizes range from 1 to 15 kWh along with different input data scenarios listed in Table 4.1 are all tested and averaged for each analysed number of cluster.

For the plot labels, the prefix "net" and "naive" is used to represent the two tested methods: the net meter clustering approach and the naive forecasting method. The suffix is used to differentiate various input data lengths (i.e. "one_season" indicates applying one season of input data to extrapolate data in other seasons). It should also be noted that Figure 4.10 illustrates errors for two types of costs, one is the yearly electricity costs before installing a battery ("pre_cost") and the other one is the costs after installing a battery ("batt_cost").

By comparing errors in yearly savings, it is clear that the net meter clustering approach outperforms the naive method for both Time-of-Use (ToU) and flat tariffs. Especially for the flat tariff case, using one month of data and the net meter clustering model produce more minor errors than applying one season of data with the naive forecasting approach for all the evaluated numbers of seasonal clusters. Another prominent trend is that as the number of seasonal clusters increases, the NRMSEs in yearly savings are reduced for all the analysed input data lengths. Moreover, when low numbers of clusters are adopted for the net meter clustering method, the errors in savings are lower for flat tariff compared to ToU however as the number of clusters increases, the NRMSE drops more quickly for ToU. As a result, they both have similar NRMSEs in estimated yearly savings at high number of seasonal clusters.

Errors in yearly costs seem to present similar trends as the errors in savings, the net meter clustering approach tends to have much smaller NRMSEs in yearly electricity costs before and after installing batteries and the differences between the net meter clustering approach and the naive method get larger when the number of seasonal clusters increases. When the net meter clustering model is applied, one month input data outperforms the naive forecasting method using one season of data for tested tariff structures and applying one season input data result in similar NRMSEs as the naive forecasting approach with two seasons of input net meter energy data.

This means by applying net meter clustering, better estimations in yearly electricity costs and battery savings can be made when a limited amount of net/gross meter data is provided. Not only this can improve the battery sizing procedures of installers or utility, but potentially it can also better assist the end-users to select the best tariff offers to reduce their energy costs with a small amount of historical data for their home energy systems. As shown in Figure 4.10, the NRMSEs in costs before and after installing a battery are both much lower using the net meter clustering approach for both evaluated tariff structures. Therefore, the solar customers could apply different tariff structures on their data extrapolated by the net meter clustering model and expect much smaller errors in estimated electricity costs compared to the baseline naive forecasting method, regardless of whether future battery purchase decisions are considered.

Another aim of the study was to explore whether the DBI is correlated to the battery sizing results. Figure 4.11 shows the errors in yearly battery saving against averaged seasonal DBI values. We can see a linear correlation between DBI and NRMSEs in yearly savings for all the evaluated tariff structures and input data lengths. This indicates that DBI can potentially be used as a metric for our end application. Hence, when a new dataset is provided, instead of going through different numbers of seasonal clusters and comparing the results, the mean seasonal DBI values can be used to directly select the best number of seasonal clusters, which heavily reduces the computational costs.

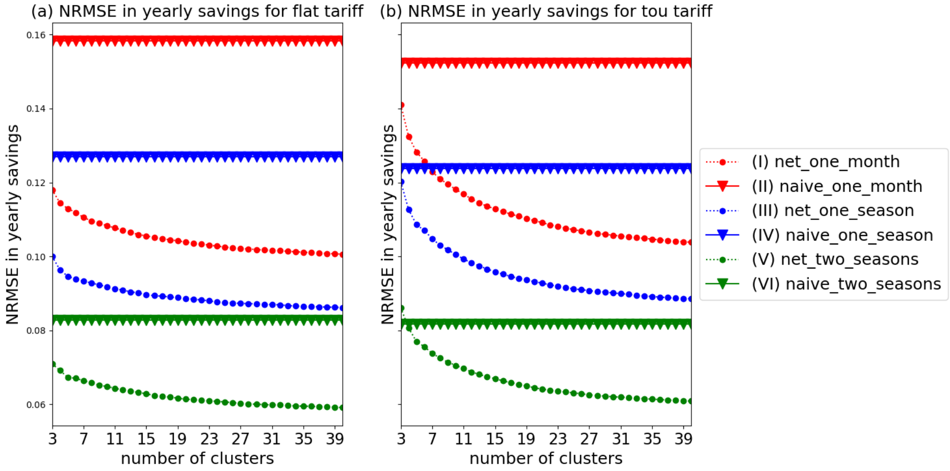

Figure 4.9: Errors in estimated yearly savings under (a) a flat and (b) a ToU tariff when applying the proposed net meter clustering method with (I) one month, (III) one season & (V) two seasons of input data and the naive forecasting method with (II) one month, (IV) one season & (VI) two seasons of input data vs number of seasonal clusters per season.

4.6.2 Errors in NPVs and Optimal Sizes

NRMSEs in NPVs at the end of a battery's lifetime against the number of seasonal clusters for both naive forecasting case and net meter clustering case with

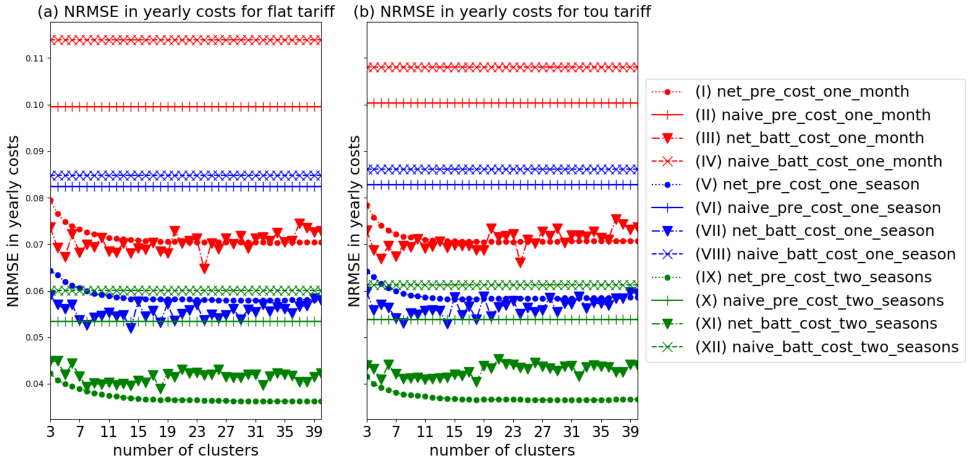

Figure 4.10: Errors in estimated yearly costs before and after a battery is installed under (a) a flat and (b) a ToU tariff when applying the proposed net meter clustering method with (I, III) one month, (V, VII) one season & (IX, XI) two seasons of input data and the naive forecasting method with (II, IV) one month, (VI, VIII) one season & (X, XII) two seasons of input data vs number of seasonal clusters per season. (note “pre_cost" and “batt_cost" respectively indicate yearly costs before and after a battery install)

different input data lengths are displayed in Figure 4.12. Again the net meter clustering method has better performances than the naive forecasting approach for almost all tested scenarios except for one case where two-season input data and three seasonal clusters are applied. The differences in NRMSEs between the two methods are extremely large when only one month of data is used to extrapolate other data in a year. As a result, this shows the net meter clustering produces much better estimations on the profitability of installing a battery system compared to the naive forecasting model. This indicates that with a small amount of gross/net meter data, the net metering clustering approach can help the customers have better ideas of whether they would make a profit or loss at the end of the battery lifetime.

Figure 4.13, Figure 4.14 and Figure 4.15 illustrate the mean true optimal battery size derived by the ideal case where all the data is provided, the mean absolute error (MAE) and r-squared value (r2) in optimal battery sizes for the net meter approach and naive forecasting method under various battery price ranges. A constant installation price of $400 is also assumed. Both tariff structures (flat and ToU) are evaluated. Forty seasonal clusters are adopted for the net meter clustering approach and average the results for all the input data scenarios in Table 4.1. The net meter clustering model outperforms the naive forecasting method in terms of MAEs and r2 values for most battery

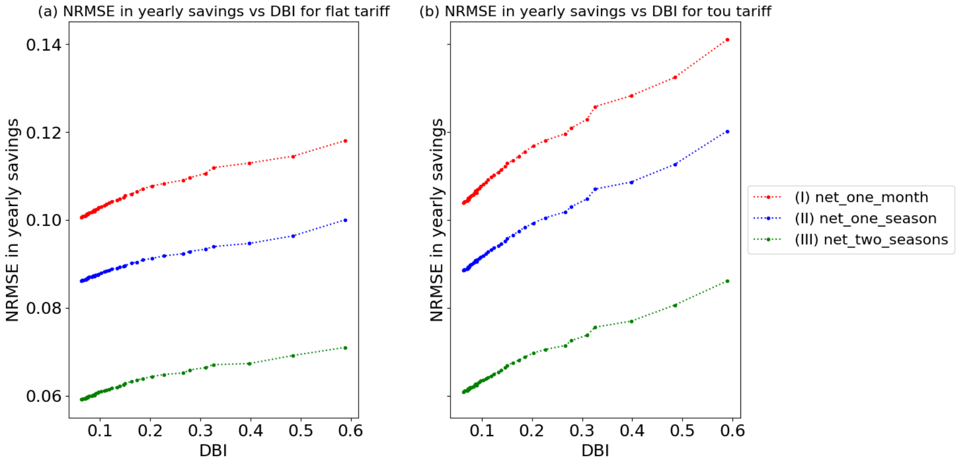

Figure 4.11: Errors in estimated yearly savings under (a) a flat and (b) a ToU tariff when applying the proposed net meter clustering method with (I) one month, (II) one season & (III) two seasons of input data vs mean seasonal DBIs

and installation cost ranges, except for the cases where the true optimal sizes are quite close to zero. For the low battery price range ($200 per kWh), the developed model achieves r-squared values of 0.72 and 0.68 using a month of input data under the specified flat and ToU tariff, which is a quite good level of accuracy.

At a lower cost range, both methods show better r2 values compared to medium battery costs. This is expected as the price increases, the optimal size tends to shift towards zero which means its variance will be much smaller compared to the residual sum of squares. Overall for the medium and large price ranges ($400-$600/kWh), the optimal battery sizes computed for ToU are larger compared to flat tariff. This means that ToU is a better option for these customers in terms of nancial returns if they decide to install a battery, as it will probably take a while for battery costs to drop to $200 per kWh.

4.7 Summary

In this chapter, a clustering analysis is performed on net meter energy data. It demonstrates that we could apply the correlations between seasonal cluster distributions to develop a battery sizing model that is quite robust to a limited amount of input net meter energy data. With a limited amount of net/gross meter energy data, by applying our proposed model with net meter energy data clustering on the test set of 262 Australian solar customers, much better

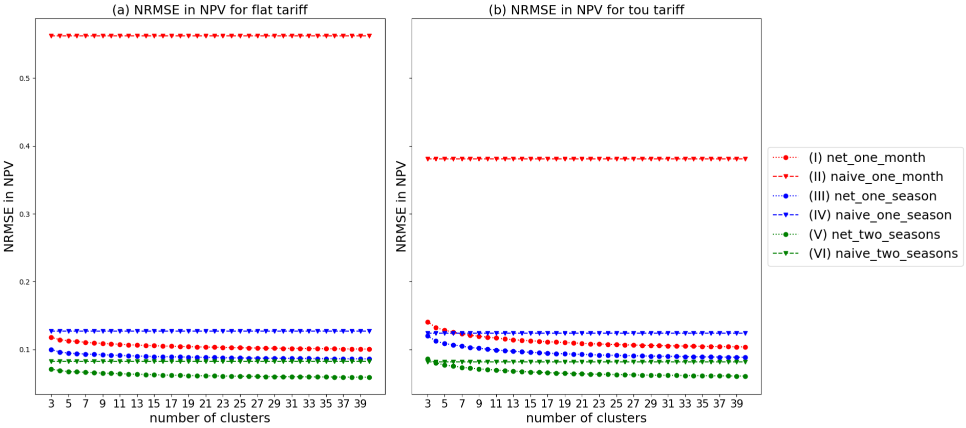

Figure 4.12: Errors in estimated end NPV under (a) a flat and (b) a ToU tariff when applying the proposed net meter clustering method with (I) one month, (III) one season & (V) two seasons of input data and the naive forecasting method with (II) one month, (IV) one season & (VI) two seasons of input data vs number of seasonal clusters.

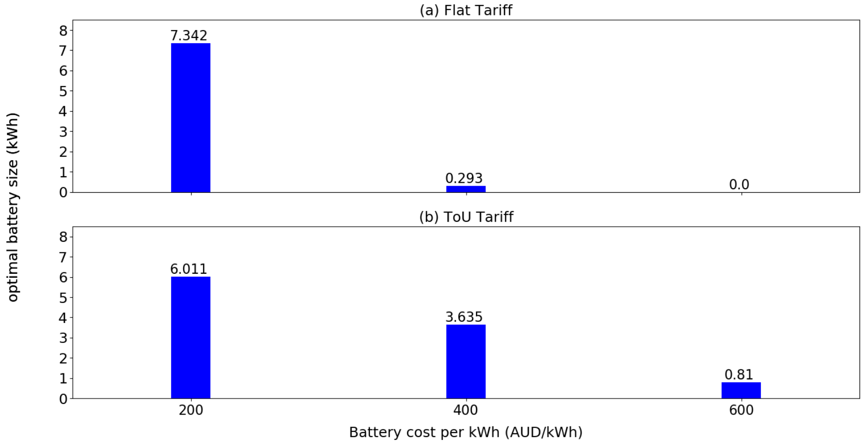

Figure 4.13: Under (a) a flat tariff or (b) a ToU tariff, the mean optimal battery size derived using a full year of data.

results have been achieved in terms of estimated annual savings (Figure 4.9), costs before and after battery installations (Figure 4.10), end NPV (Figure 4.12) and optimal sizes (Figure 4.14 and Figure 4.15) compared to the baseline naive

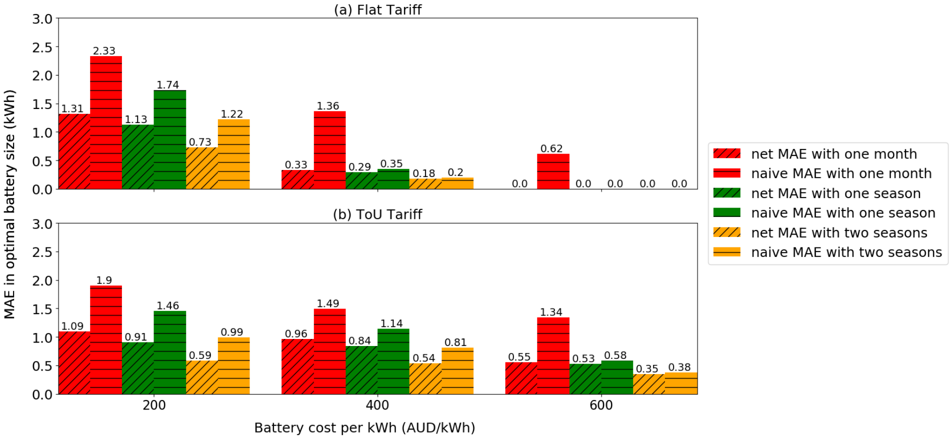

Figure 4.14: Under (a) a flat tariff or (b) a ToU tariff, the MAE in estimated optimal battery sizes using the net meter clustering approach and naive forecasting method for different battery price ranges and input data lengths.

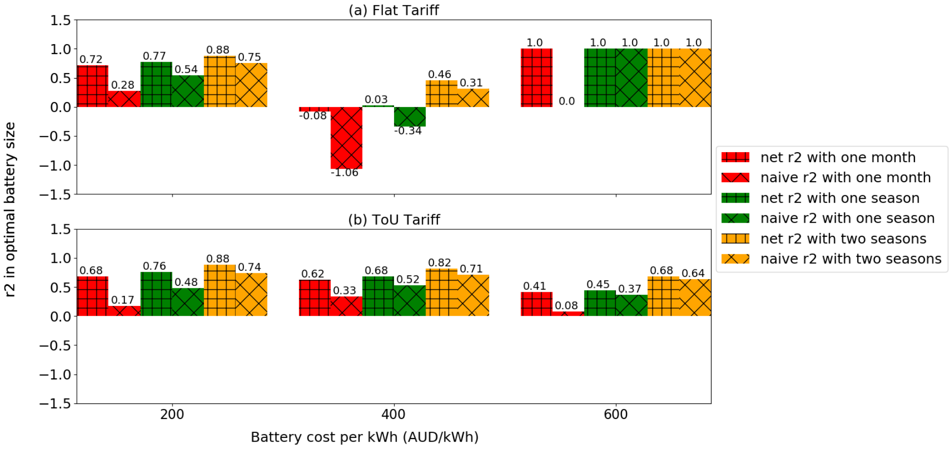

Figure 4.15: Under (a) a flat tariff or (b) a ToU tariff, the mean R-squared value in estimated optimal battery sizes using the net meter clustering approach and naive forecasting method for different battery price ranges and input data lengths.

forecasting approach.

For end-users who do not have easy access to enough historical smart meter data, the net metering clustering approach could be used to predict their annual electricity costs and battery profitability under different tariff structures. As a result, the proposed model could be implemented as a feature of a home energy recommendation tool to help residential customers make better tariff selection and battery purchase decisions with loose requirements on the length and quality of the input data. Moreover, installers and utilities which are likely to deal with customers with insufficient net meter data during the ongoing net meter rollouts could utilise this technique as a recommendation service for their customers. They could also gain valuable insights into the impacts of tariff offers and battery prices on the electricity bills of their customers and make better predictions of the solar/battery market trends with a small amount of net/gross meter energy data.

Chapter 5

Synthetic PV and Load Data Generation via Generative Adversarial Networks

5.1 Introduction

In Chapter 4, a data interpolation model is introduced which could produce a year worth of PV and consumption energy data from one month of input data and achieve results with satisfactory accuracy in a battery sizing model. However, in the absence of any measured historical data, the above approach cannot function. As a result, synthetically generated data can be used to model the data distributions and generate possible trajectories of PV and load power. It can be used in an end-use application such as a battery sizing model.

As an active research field in machine learning, GANs have been widely explored and applied in computer vision. The main advantage of GANs is its ability to generating high-quality samples with less statistical assumptions and faster runtime compared to other generative approaches such as Markov chains Monte Carlo methods or variational autoencoder [186].

This chapter introduces a DCGAN based model to generate synthetic PV generation and load power data under various data synthesis scenarios. It presents a detailed analysis, including comprehensive evaluations and alternative approaches that address the above shortcomings observed in the relevant literature.

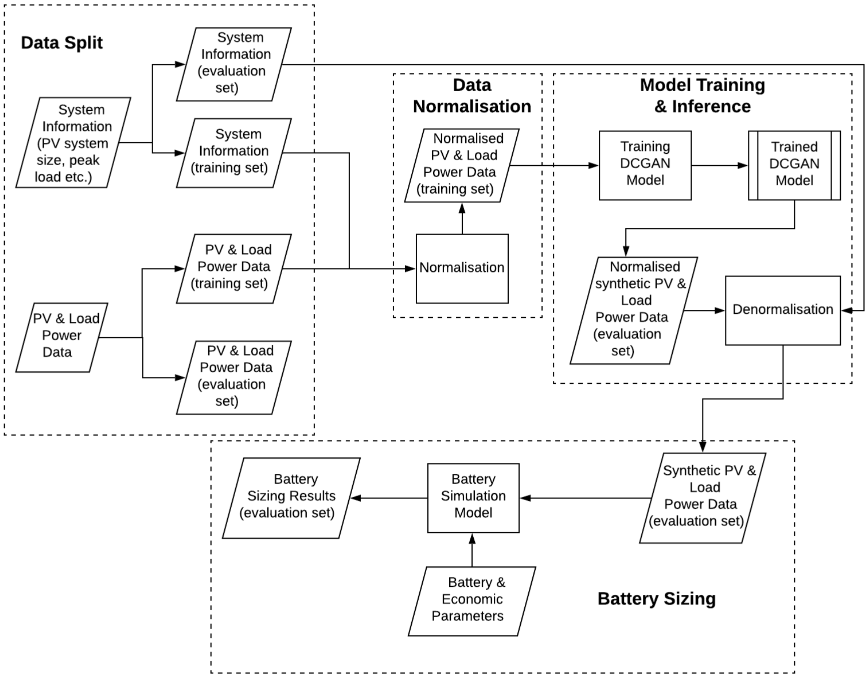
Data Split
System Information (PV system size, peak load etc.)
System Information (evaluation set)
System Information (training set)
PV & Load Power Data
PV & Load Power Data (training set)
PV & Load Power Data (evaluation set)
Data Normalisation
Normalised PV & Load Power Data (training set)
Normalisation
Model Training & Inference
Training DCGAN Model
Trained DCGAN Model
Normalised synthetic PV & Load Power Data (evaluation set)
Denormalisation
Synthetic PV & Load Power Data (evaluation set)
Battery Simulation Model
Battery Sizing Results (evaluation set)
Battery & Economic Parameters
Battery Sizing

Figure 5.1: The flowchart to perform data synthesis and battery sizing using the generated synthetic data.

5.2 Data Synthesis using Deep Convolutional Generative Adversarial Networks

This chapter aims to synthesise daily residential PV and load profiles using a DCGAN framework, then validate the synthetic data with respect to an end-use application that performs battery sizing for new customers without historical data. As shown in Figure 5.1, the adopted dataset is first separated into a training set and an evaluation set. System information is applied to normalise the training and evaluation sets. The normalised training data is used to train a DCGAN model. By inferencing the trained model, synthetic data is generated and fed into the developed residential battery sizing tool to estimate electricity costs and optimal battery sizes for the PV households in the evaluation set.

An alternative copula approach (described in Section 5.2.3) and a comparative constant normalised profile approach (described in Section 5.2.4) are also implemented for all the considered data synthesis scenarios and applications to allow a comprehensive comparison. To properly assess the errors in the three modelling approaches concerning the end-use application, the battery sizing results are also computed for the households in the evaluation set with the ideal case where a whole year of real-time PV and load data is provided instead of synthetic data.

5.2.1 Data Collection and Preprocessing

Dataset

The dataset collected by Solar Analytics [160] is also used in this study. This includes one year 5-minute average PV generation and load power data of 2925 Australian residential solar households, collected between 1st January 2017 and 31st December 2017. For simplicity, in this chapter, we also refer to the 5-minute average PV generation and load power data as PV and load data. The rated DC powers of these rooftop PV systems were also recorded. This chapter also considers the net meter power data synthesis for households with net metering schemes where solar generation is first used on-site and the excess generation is then exported. Eqn. 5.1 is used to convert 5-minute PV and load average power to 5-minute net meter average power data:

$$net_{power} = PV_{power} - load_{power} \tag{5.1}$$

As one of the aims of this work is to conduct a concrete validation of the proposed model, a large dataset is preferred for training and evaluation. This however, does not mean a large dataset is necessary for the proposed model to function. Exploring how much data is required for the model to perform properly will be interesting however, it is not within the scope of this thesis.

Data Split

The dataset is divided into a training set, a validation set, and an evaluation set to properly validate the proposed DCGAN framework and residential battery sizing tool. The training set which contains data from 80% of the total households is used to fit the DCGAN model and the alternative approaches, on the other hand, the 10% of the households are used in the validation set with the purpose of selecting the optimal checkpoint of the DCGAN model and to select the appropriate copula function for the copula approach (more details are described in Section 5.2.2 and Section 5.2.3). Then the remaining 10% of the residential households are treated as new customers where it is assumed that they have no historical meter data and their information is not used during the training processes.

Data Normalisation

To accelerate the training process and achieve faster convergence for both copula and DCGAN approaches, we normalise the training data to a small numerical range that matches the intended output ranges of these models. As the intended targeted user group of the synthetic data is new customers who have no meter data or survey data, it is desirable to use as little information as possible to generate synthetic data. Hence the DC PV rating and peak load are used for normalisation and denormalisation as they are fairly accessible in practice. For each household in our dataset, PV generation data is normalised by the DC power rating of the PV system; load power data is normalised by the peak 5-minute average load power and the net meter power data is normalised by the higher values of peak 5-minute average load power and DC PV power rating. This ensures the normalised values of PV and load are within the range of [0, 1] and the normalised net meter power is within [-1, 1]. These ratings are also applied to denormalise the synthetic outputs from the evaluated generative approaches.

5.2.2 Synthetic Data Generation by DCGAN

In this subsection, the original GANs and DCGAN models are firstly introduced, then we describe the proposed *base* DCGAN model with several changes compared to the original DCGAN approach [156]. These make the model more suitable for generating PV and load power data. A few different data synthesis scenarios are then considered with various modifications on the architecture of the base model, as demonstrated in Figure 5.2. Depending on whether PV and load data are generated separately or simultaneously, the base DCGAN model is adjusted to a single-channel or double-channel DCGAN model. A single-channel DCGAN model generates synthetic PV/load/net meter power data separately. In contrast, a double-channel DCGAN model outputs both PV and load daily profiles at the same time. Moreover, suppose additional labels of PV/load/net meter pro- les are provided during the training and inferencing process. In that case, the base model is converted to a conditional single-channel or double-channel DCGAN model.

Generative Adversarial Networks

A GAN structure is illustrated in Figure 5.3 which has two main functions that are both differentiable and typically are implemented by artificial neural networks: a generator (G) and a discriminator (D). The generator function takes latent noise ($\mathbf{z}$) sampled from a simple prior distribution $p_{\mathbf{z}}$ (e.g. a Gaussian distribution) and outputs synthetic data ($G(\mathbf{z})$). Then real and synthetic data are fed into the discriminator, and it outputs the probability of inputs being real data and assigns real/fake labels to the input samples. The goal for the discriminator is to minimise the cross-entropy cost function J_D defined in Eqn. 5.2 only by adjusting its parameters θ_D hence maximises the probability of assigning the correct labels to real samples ($\mathbf{x}$) and synthetic samples ($G(\mathbf{z})$):

$$J_D(\theta_D, \theta_G) = -\frac{1}{2}\mathbb{E}_{\mathbf{x}\sim p_{data}}[\log D(\mathbf{x})] - \frac{1}{2}\mathbb{E}_{\mathbf{z}} \sim p_{\mathbf{z}}[\log(1 - D(G(\mathbf{z})))] \tag{5.2}$$

Where θ_D, θ_G are the parameters of the discriminator and generator, p_{data} is the real data distribution.

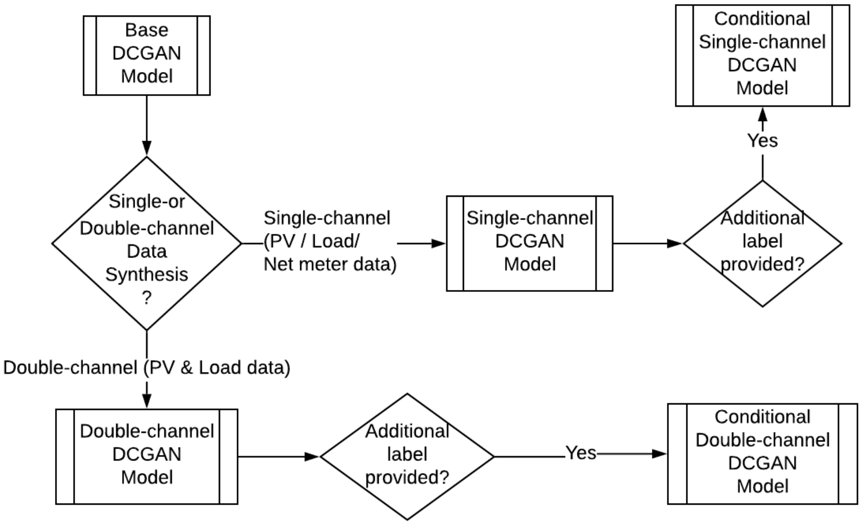

Figure 5.2: The base DCGAN model and its variations designed for different data synthesis scenarios.

On the other hand, the generator tries to maximise the probability that the discriminator is mistaken by minimising the cost function in Eqn. 5.3 only by adjusting its parameters θ_G:

$$J_G(\theta_D, \theta_G) = -\frac{1}{2}\mathbb{E}_{\mathbf{z}} \sim p_{\mathbf{z}}[\log(D(G(\mathbf{z})))] \tag{5.3}$$

As J_G is inversely correlated to J_D, it is possible to combine the cost functions of D and G to form a min-max two-player game between D and G with a value function defined in Eqn. 5.4:

$$\min_G \max_D V(D, G) = \mathbb{E}_{\mathbf{x} \sim p_{data}}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}} \sim p_{\mathbf{z}}[\log(1 - D(G(\mathbf{z})))] \tag{5.4}$$

During the training process, a minibatch of real data is sampled from the training set and another minibatch of $\mathbf{z}$ is sampled from $p_{\mathbf{z}}$ for each training step. Two gradient updates are performed simultaneously: one to update D to maximise $V(D, G)$, the other one to update G to reduce $V(D, G)$. It is mathematically proven in [155] that when both D and G are at their local optimum, the min-max game reaches the Nash equilibrium and GANs converge. As a result, the data distribution of $G(\mathbf{z})$ is same as the training data and D outputs 50% real probability for both real and synthetic samples.

Deep Convolutional Generative Adversarial Networks

By successfully utilising multiple convolutional and deconvolutional layers [187] for both generator and discriminator, DCGAN [156] can generate high-quality

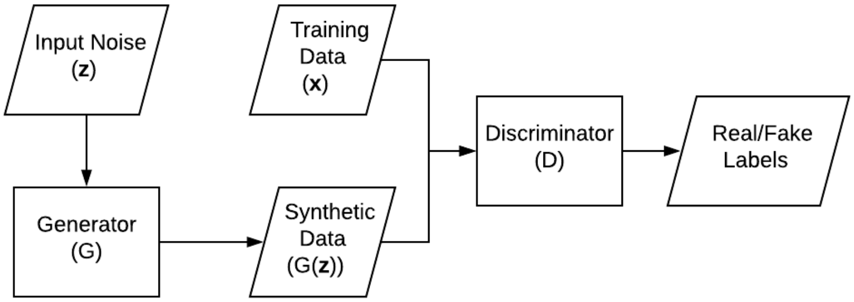

Figure 5.3: The main framework of GANs [155].

samples efficiently and as a result it forms the basis of many recent proposed GANs models.

A convolutional layer [188] is similar to a fully-connected layer (sometimes referred as a dense layer) in an artificial neural network, they both contain neurons (sometimes referred as nodes). Each neuron receives inputs and applies dot products to inputs using a set of learnable weights, followed by an activation function that is a fixed mathematical function to provide non-linearity to the networks.

The main difference between a dense and a convolutional layer is how their neurons and connections are arranged. In a dense layer, its input is a vector and each of its neurons are independently connected to all neurons in the previous layer. In contrast, a convolutional layer generally has a three-dimensional (3D) input with neurons also arranged in a 3D manner and connected only to a small region of the previous layer. This region is referred to as a receptive field and its corresponding array of weights is called a filter. The dimensions of neurons in a convolutional layer are defined as height, width and depth. Figure 5.4 shows the process of applying a single filter to the input of a convolutional layer, where the depth of the input is set to 1 for a more accessible demonstration in two dimensions (height × width). Padding is in Figure 5.4, as the boundary zero values added prior to applying the filters. This preserves the size of the input volume when multiple convolutional layers are used. The same filter slides multiple times across the height and width of the input matrix and performs dot products on the receptive fields to output a feature map. Typically multiple filters are used in parallel, hence multiple feature maps are generated and stacked to create the output volume of a convolutional layer. The stride value which is distance between two consecutive receptive fields, along with the padding amount, the filter size and the number of filters, are the four main hyperparameters in a convolutional layer that control the size of the output volume. A deconvolutional layer, sometimes also referred to as a transposed convolutional layer, reverts the spatial transformation of a convolutional layer.

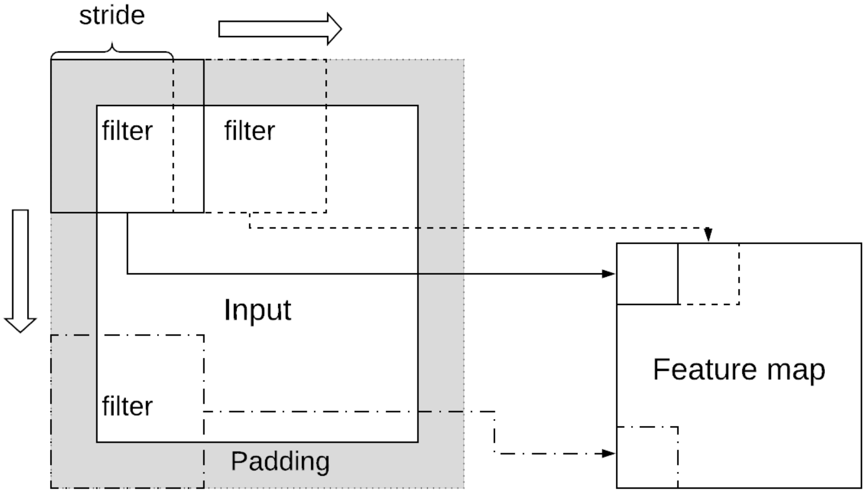

Figure 5.4: The process of applying a filter in a convolutional layer

By using partially connected neurons, convolutional/deconvolutional layers significantly reduce the model parameters, which reduces the computational costs and the risk of over-fitting within an artificial neural network. This is quite helpful especially when dealing with high-resolution samples such as images or time series. Moreover, by applying various filters and stacking multiple convolutional/deconvolutional layers, spatial dependencies in images and temporal dependencies in time series data can be captured without extra data preprocessing steps than just normalisation of the input data.

Despite the success of the original DCGAN model in modelling the real data distribution, it could lead to some artifacts in the individual generated profiles which lead to noticeable differences between the synthetic and real daily power profiles. In the synthetic clear-sky profiles generated by the original DCGAN approach, often there are some disturbances in the power curve whereas a real clear-sky profile is a smooth bell-shape curve. Often in load profiles with low night-time/early morning load, base load with regular cycles (e.g. refrigeration cycles) can be visually identified however these consistent base load cycles are not observable in the synthetic profiles generated by our previous model which affects the validity of the model on an individual household's level. These artifacts are demonstrated in Figure 5.5, where real profiles and the synthetic profiles generated by the modified DCGAN model introduced in this study are also shown. A clear-sky PV profile and a base load profile are picked from the evaluation set with real-time data and then we find their closest matching PV and load profiles in terms of Euclidean distances in the synthetic evaluation sets generated by the previous and the modified DCGAN models. It is suspected that these artifacts are caused by using deconvolutional layers in the generator [189]. As a deconvolutional layer conducts an inverse version of the spatial transformation shown in Figure 5.4, when filter sizes are not divisible by strides, it leads to uneven overlaps in some of the outputs of the deconvolutional layer. This issue is described in details in [189], where the authors suggest applying nearest-neighbor interpolation to up-sample the input of each layer (except for the output layer of the generator) and adopt convolution layers instead of deconvolutional layers to avoid generating these artifacts. Overall, to deal with the artifacts and generate samples with better quality, a few changes have been implemented on the model architecture of the original DCGAN approach:

1. Adopt the resize-convolution approach suggested in [189] instead of deconvolutional layers in the generator to reduce the artifacts in synthetic samples.

2. These artifacts tend to often occur around the boundary of the output matrix of the generator. To further reduce these artifacts, the amount of padding is increased and strides are reduced from 2 to 1 for the first two convolutional layers of the discriminator to make it easier for the discriminator to identify these boundary artifacts.

3. Instead of adopting the same model architecture for PV and load power data synthesis, different model structures are applied for various data synthesis scenarios. Empirically, more convolutional filters in the generator result in better results for generating synthetic load data than PV data.

As a result, as illustrated in Figure 5.5, now the synthetic profiles generated by the new DCGAN model look more realistic and almost make no visual difference to the real power curves without just memorising the profiles in the training data.

Single-Channel/Double-Channel Data Synthesis

To fully utilise the structures of convolutional layers, a single day of 5-minute average power data is shaped to a three dimensional tensor with dimensions of $rows \times columns \times channels$. Each row contains data for 24 consecutive 5 minute timestamps and each channel as a data channel of power data. As a result, one day of 288 5-minute average power values for a single data source is represented by a tensor of 24 *rows* × 12 *columns* × 1 *channel*. For double-channels data synthesis, PV & load data for a single day is converted into two channels: PV power data and the other for load power data.

The single-channel DCGAN architecture for generating PV power data is shown in Figure 5.6. To generate a single sample, the input to the generator is a vector of 100 random noises sample from a Gaussian distribution of $\mathcal{N}(\mu, \sigma^2)$ with $\mu = 0$ and $\sigma = 1$. After going through a full-connected neural network layer and a few layers of convolutional layers, the generator produces a three-dimensional output tensor that is flattened to a synthetic daily power profile.

.

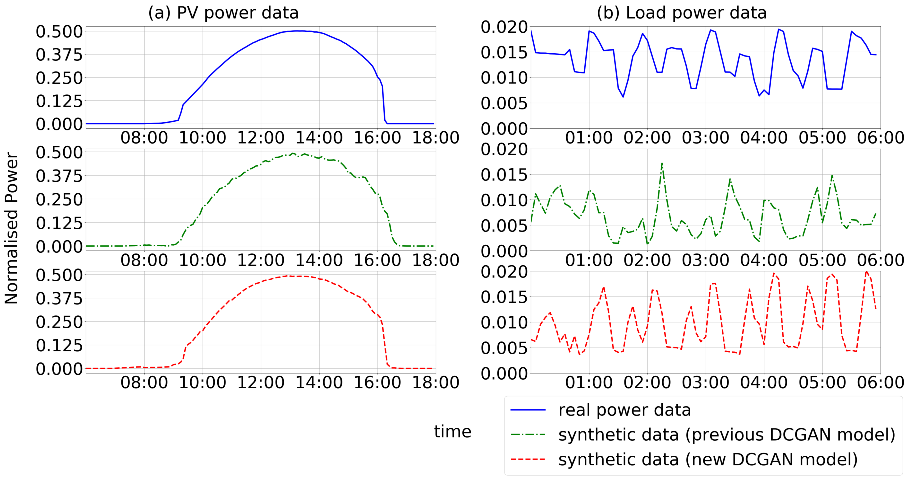

Figure 5.5: Comparisons of daily (a) PV profile for a clear-sky day and (b) a load profile's distinguishable base load in early morning between real data, synthetic data generated by our previous approach [22] and the model in this paper.

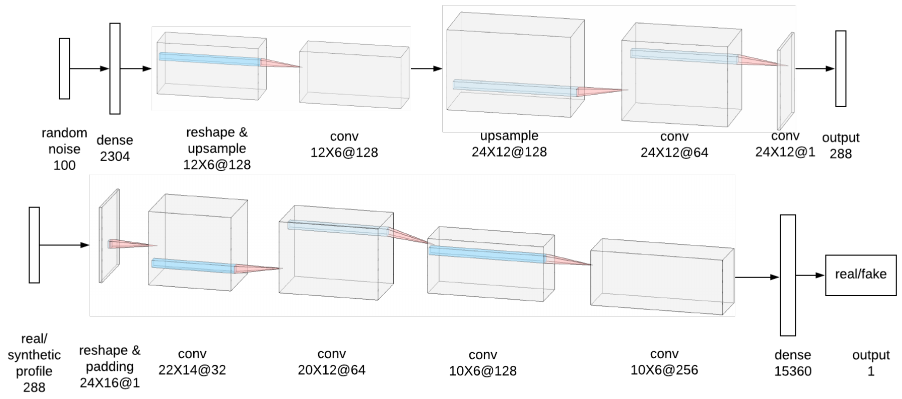

Figure 5.6: DCGAN Architecture used for single-channel PV data synthesis for the generator (top row) and the discriminator (bottom row). Numbers at the bottom of each layer indicate the output dimensions of each layer. Layer type "conv" refers to a convolutional layer. Layer type "upsample" refers to an upsampling layer using nearest-neighbor interpolation which repeats the rows and columns of the input matrix by two times and layer type "dense" refers to a dense layer. A single number describes a dense layer or input/output vector dimension whereas $rows \times columns@channels$ shows dimensions of a 3D output.

Table 5.1 illustrates the model architectures adopted for various data synthesis scenarios, with dimensions of inputs, outputs of the generator and discriminator and the hyperparameters of each neural network layer. Two-dimensional (2D) convolutional layers are adopted for both the generator and the discriminator. The reason for not using a one-dimensional (1D) architecture for the convolutional layers is that it actually ends up with slightly worse results compared to using a 2D architecture. To generate a single sample, the input to the generator is a vector of 100 random noises sample from a Gaussian distribution of $\mathcal{N}(\mu, \sigma^2)$ with $\mu = 0$ and $\sigma = 1$. Then after going through a full-connected neural network layer and a few layers of convolutional neural network layers, the generator produces a three-dimensional output tensor which is flattened to a synthetic daily power profile. Similar to the model guidelines proposed in the original DCGAN paper, batch normalisation [190] is applied for each convolutional layer, except for the last year of the generator and the first layer of discriminator. ReLU [191] and LReLU [192] activation functions are applied for both the generator and the discriminator layers except for their output layers. For the output layer of the discriminator, the activation function is sigmoid however different generator output activation functions are adopted for different data synthesis scenarios in Figure 5.2 with regards to their normalised numeric ranges: sigmoid for PV or load / PV & load, hyperbolic tangent (tanh) for net meter power. An Adam optimiser [193] with a learning rate of 0.0002 and momentum β_1 of 0.5 is applied and the batch size is set to be 128. Dropouts [194] are applied for each convolutional layer in the discriminator to prevent over-fitting and the rate is set to be 0.25.

Conditional Synthetic Data Generation

The DCGAN model is unconditional, which means we do not control the samples it generates. On the other hand, if some additional information is available, it is possible to utilise the information to guide the data synthesis process via a conditional GAN model. Conditional GAN (CGAN) was first proposed in [195] where they demonstrate the image synthesis conditioned on class labels and it is also reported in [196] where they found that by using class-conditional GAN, much better samples are generated compared to an unconditional GAN model. It is still not clear how exactly the provided auxiliary information improves the data generation process, one common hypothesis is that the extra information provides useful features to the generator and the discriminator during the training process [186]. Inspired by the CGAN structure, we convert the DCGAN model to a conditional DCGAN model. The aim is not only to demonstrate that we could use DCGAN generate PV and load data conditioned on additional information, but also to explore whether providing auxiliary information could improve the results and what information is useful with regards to end-use applications such as estimating electricity costs and optimal battery sizes for new customers.

CGAN requires minor modification from a standard GAN where we only need to make sure we provide the extra information to both discriminator and

Table 5.1: The model architectures for different data synthesis scenarios. Layer type "conv" refers to a convolutional layer with its specifications (kernel size (k), number of filters (n) and stride (s)). Layer type "upsample" refers to an upsampling layer which repeat the rows and columns of the input matrix by two times and layer type "dense" refers to a full-connected neural network layer with its corresponding number of hidden units. For input and output dimensions, a single number of input/output represents a vector with its length and three numbers indicate a three-dimensional tensor with the format of rows × columns × channels.

Data Synthesis Type			
PV	Load	Net	PV & Load
Layer Type, Specifications			
Generator			
input, 100	input, 100	input, 100	input, 100
dense, 2304	dense, 2304	dense, 2304	dense, 2304
upsample	upsample	upsample	upsample
conv, k3n128s1	conv, k5n256s1	conv, k5n256s1	conv, k3n128s1
upsample	upsample	upsample	upsample
conv, k3n64s1	conv, k5n128s1	conv, k5n128s1	conv, k3n64s1
conv, k3n1s1	conv, k3n1s1	conv, k3n1s1	conv, k3n32s1
			conv, k3n16s1
			conv, k3n2s1
output, 24×12×1	output, 24×12×1	output, 24×12×1	output, 24×12×2
Discriminator			
input, 24×12×1	input, 24×12×1	input, 24×12×1	input, 24×12×2
conv, k3n32s1	conv, k3n32s1	conv, k3n32s1	conv, k3n32s1
conv, k3n64s1	conv, k3n64s1	conv, k3n64s1	conv, k3n64s1
conv, k3n128s2	conv, k3n128s2	conv, k3n128s2	conv, k3n128s2
conv, k3n256s1	conv, k3n256s1	conv, k3n256s1	conv, k3n256s1
dense, 15360	dense, 15360	dense, 15360	dense, 15360
output, 1	output, 1	output, 1	output, 1

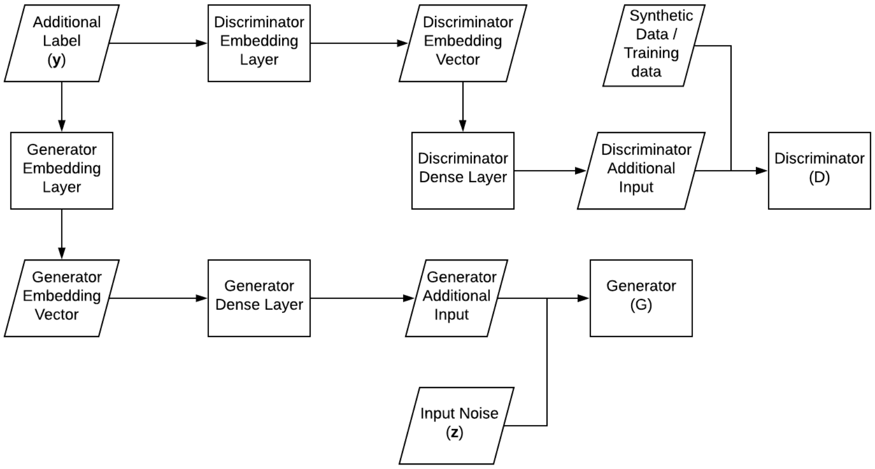
Additional Label (y)
Discriminator Embedding Layer
Discriminator Embedding Vector
Synthetic Data / Training data
Generator Embedding Layer
Discriminator Dense Layer
Discriminator Additional Input
Discriminator (D)
Generator Embedding Vector
Generator Dense Layer
Generator Additional Input
Generator (G)
Input Noise (z)

Figure 5.7: The structure of the input layers for the conditional DCGAN model.

generator as an extra input layer. Consider the additional information as $\mathbf{y}$, the value function of CGAN can be extended to Eqn. 5.5 from Eqn. 5.4:

$$\min_{G}\max_{D} V(D,G) = \mathbb{E}_{\mathbf{x}\sim p_{data}}[\log D(\mathbf{x}|\mathbf{y})] + \mathbb{E}_{\mathbf{z}} \sim p_{\mathbf{z}}[\log(1 - D(G(\mathbf{z}|\mathbf{y})))] \quad (5.5)$$

CGAN is quite flexible in how the extra information is represented (by one-hot encoding or embedding [197]) and where this information is inserted, as the position can be any of the stacked layers within the GAN. For simplicity, in this chapter, a single class label is used and the embedding representation [197] of the label is applied for the developed conditional DCGAN model. Apart from the additional inputs containing class information, the same model architectures in Table 5.1 are applied for each data synthesis scenario. Figure 5.7 shows how the additional class label is converted and concatenated within the conditional DCGAN model: class label $\mathbf{y}$ is first passed to an embedding layer to generate an embedding vector then it gets passed to a dense layer. It should be noted that both the embedding layer and the dense layer are trained along other layers of the generator and the discriminator according to the loss functions of both players. After passing through the dense layer, additional input vectors for both players are concatenated along with their default inputs: for the generator, the concatenation is simply to join two vectors together whereas for the discriminator, the additional input vector is reshaped to form an additional channel to the input synthetic/real data.

We use the month number as the class label for all the data synthesis scenarios in Figure 5.2 as seasonality exists for both PV & load profiles and more importantly, it can be retrieved directly from timestamps without any extra information from households. Another label we consider for load synthesis is an arbitrary label called *week-day-focus*, which is determined by whether the daily profile belongs to a weekday or weekend and whether the household has a *day-focus* consumption pattern. The day-focus pattern is simply decided by whether annually the household has more consumption between 8:00 am to 4:00 pm compared to its consumption between 4:00 pm and 12:00 am during weekdays/weekends. On the other side, if it is the opposite situation the household is considered to have an *evening-focus* load pattern. As a result, there are in total four possible values for the week-day-focus class label: 0 for a weekday with day-focus consumption pattern; 1 for a weekday with evening-focus pattern; 2 for a weekend with day-focus consumption pattern; 3 for a weekend with evening-focus pattern. It should be noted that potentially It should be noted that potentially many dierent types of information regarding the power proles/households could be used as class labels and multiple labels could be embedded together to synthesise power proles. However, for the scope of this thesis we only consider applying either month number or week-day-focus as profile label to avoid creating too many scenarios and make sure the label information is easy to retrieve.

Checkpoint Selection using validation set

To deal with the oscillatory performance issue of the GANs described in [186] and [198], the proposed DCGAN model is trained for a sufficient amount of time until the losses of the generator and discriminator stabilise and achieve the Nash equilibrium. The parameters of the trained model are saved for every 50000 training steps. The saved parameters of the model after a given training step is also referred to as a checkpoint. This allows the model to generate synthetic data and evaluate different checkpoints without re-training the model. Eventually, the checkpoint with the validation set's best performance is selected as the checkpoint adopted for the DCGAN model. The metric used to select the optimal checkpoint is the normalised root mean squared error (NRMSE) used for calculating electricity costs under a flat tariff structure by comparing electricity costs of synthetic data and real data for households in the validation set. This metric is a good indication of the errors in the energy imports and exports of households and ties well for the end-use applications of battery sizing.

5.2.3 Alternative Copula Model

As mentioned in Section 1.1, Markov chain models [13, 14] were used to generate synthetic PV power data. However, their approaches require solar irradiance data to generate synthetic clear sky index data. Unfortunately, such data was not available for the adopted dataset. Hence, instead the copula model is used as a comparative approach. A copula is a multivariate probability distribution with uniform marginal probability distributions. It can characterise the dependence of random variables independently of the marginal distribution functions of those variables [199]. For synthesising power profiles, the random variables can be described by $(x_1, x_2, x_3 ... x_n)$ where each variable represents the power value of a single 5 minute timestamp of a day and hence $n = 288$ for a single day of power profile. Suppose $F(x_1, x_2, x_3...x_n)$ is the joint cumulative distribution function (CDF) of $x_1, x_2, x_3...x_n$ and the marginal CDF of a variable x_i is $F_i(x_i)$, the Sklar's Theorem states that there exists a copula C such that:

$$F(x_1, x_2, x_3...x_n) = C[F_1(x_1), F_2(x_2)...F_n(x_n)] \tag{5.6}$$

The copula C is unique if all the marginal CDFs are continuous. Assuming all the marginal CDFs are probabilistically calibrated which leads to $u_i = F_i(x_i)$ and $F_i^{-1}(u_i) = x_i$ where $u_i \in Unif[0, 1]$, the copula C can be described by Eqn. 5.7:

$$C(u_1, ..., u_n) = F(F_1^{-1}(u_1), ..., F_n^{-1}(u_n)) \tag{5.7}$$

According to Eqn. 5.7, to construct the joint distribution, we just need to get the inverse functions of the random variable marginal distributions and then pick a copula function to model the dependence among various random variables. There are a few commonly-used copula functions such as Gaussian copula, Clayton copula, Frank copula etc. In this study, four different copula functions (Gaussian, Clayton, Frank, Gumbel) are tested using the same metric and procedure described in Section 5.2.2. As a result, we have adopted the Gaussian copula which produces the lowest NRMSE in electricity costs for all the synthesis scenarios. In this case, Eqn. 5.7 is converted to:

$$C(u_1, ..., u_n) = \Phi_{n,R}(\Phi^{-1}(u_1), ..., \Phi^{-1}(u_n)) \tag{5.8}$$

Where $\Phi_{n,R}$ is a multivariate normal distribution with zero mean and a $n \times n$ correlation matrix R and Φ^{-1} is the inverse CDF of a univariate normal distribution. To build the copula from our training data, all we need to do is to convert our empirical data of random variables to uniform using a kernel estimator of the CDF and fit the Gaussian copula [145]. Then we can sample from the fitted copula and transform the random samples back to the original scale.

The copula package in R ([200]) is used to fit the Gaussian copula and generates synthetic samples. Same as the DCGAN data synthesis described in Section 5.2.2, we also consider the data synthesis scenarios in Figure 5.2 for the copula approach. For the conditional data synthesis scenario, instead of feeding in the class label like what has been done for the conditional DCGAN model, we divide the training data conditioned on the class labels and fit separate copula functions for each class as the copula approach does not have the exact mechanism as the GANs which allows extra information during the fitting process.

5.2.4 Residential Battery Sizing Model

After synthetic power data is generated, it is fed into a residential battery sizing model where estimated annual electricity costs and optimal battery sizes are determined. The battery sizing results for using a full year of real or synthetic PV and load data are computed for households in the evaluation set , allowing us to estimate the errors for using synthetic data.

Comparative Constant Normalised Profile Approach

In addition to the alternative copula approach, the developed DCGAN approach is also compared to a simple approach commonly used in practice to size batteries for new customers with no historical meter data. The main idea of this industrial approach is to derive averaged normalised PV and load profiles from the existing PV/load dataset across a whole year. Then the mean normalised PV/load profiles are assumed to stay constant for different new customers within a specified time window (a month/a year) or a load pattern, hence in this study we refer to this method as the *constant normalised profile* approach. After obtaining a whole year of normalised PV and load data, the constant profiles are unnormalised and then fed into a battery simulation model to produce battery sizing results. This method has been adopted for most residential battery calculators (e.g. Ref [201]) with different input requirements or geographical scope. To make a fair comparison in terms of sizing results, we assume the amount of inputs are consistent for the DCGAN, copula and the constant normalised profile approaches. As a result, most of the data synthesis scenarios in Figure 5.2 are implemented for the constant normalised profile method except for double-channel synthesis as it is not feasible for this approach. The unconditional synthesis is relatively straight-forward where the training set is averaged to produce the mean normalised PV/load/net meter daily profiles. On the other hand for conditional data synthesis, profiles with the same class label are averaged which results in different mean profiles, one for each month and one for each week-day-focus label.

Battery Simulation Model

The key parameters of the battery simulation model are taken from Table 4.4, where the battery specifications, economic parameters and tariffs are listed for Australian residential solar households. We assume the battery follows the same rule-based control scheme in Algorithm 1. The battery charges when there is excess solar energy and discharges until depleted when load consumption is higher than PV generation. This control algorithm aims to maximise the onsite PV self-consumption and has been adopted in various studies (such as [158, 202]) and many installed battery systems due to its simplicity and ease of implementation.

To compute the battery sizing results, we follow the same method and assumptions used in Chapter 4 where a grid search is applied to the battery size range of 0-15kWh in order to find the optimal battery sizes in terms of the highest Net Present Value (NPV) at the end of the battery lifetime which is assumed to be 15 years. We assume a fixed installation cost of 200 Australian Dollar (AUD) and a linear battery price of 200 AUD/kWh, this linear battery price is derived from the lithium-ion battery prices predicted in [203], where the price in 2018 is 176 USD/kWh and according to the displayed trend, the price at the end of 2019 will be close to 140 USD/kWh which is equivalent to around 200 AUD/kWh. The battery savings are calculated by taking a difference between

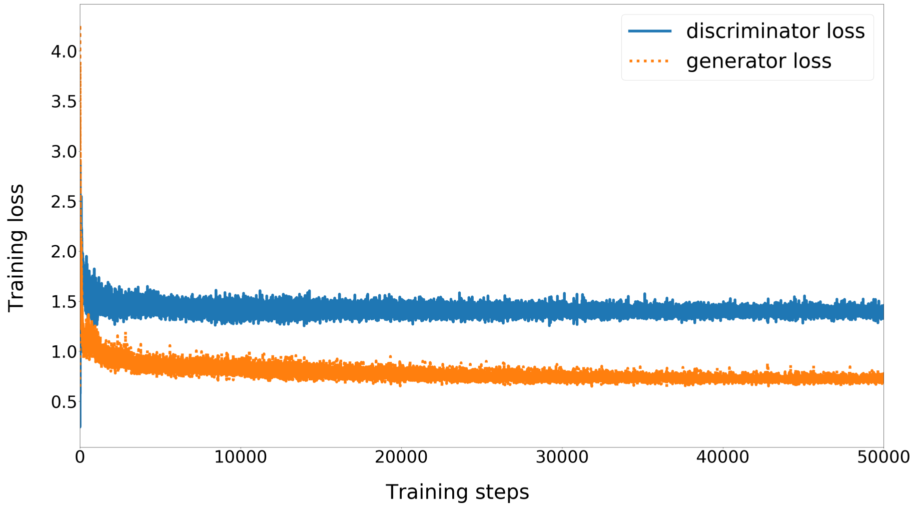

Figure 5.8: Losses of the discriminator and the generator for the first 50000 training steps when training DCGAN on PV power data

the electricity costs before and after installing battery systems. The impact of battery degradation on battery saving is quantitatively taken into account by assuming a 5% annual reduction in electricity cost savings due to battery degradation [204].

5.3 Results and Discussion

5.3.1 Model Training & Computational costs

Training of the DCGAN approach is performed on a desktop with an Intel Core i7-8700K CPU, 32 GB of RAM and an Nvidia GeForce RTX 2070 GPU, Python codes implemented using both Tensorflow [205] and Keras [206] as the machine learning libraries. As shown in Figure 5.8, which illustrates the losses of the generator and the discriminator for the first 50000 training steps during the DCGAN training for PV power data. Both the discriminator and generator losses are relatively high at the start then they drop and stabilise after approximately 40000 training steps where also the generated samples have reached a good quality judging by visual inspections. Then both losses oscillate within a small range where the Nash equilibrium is achieved. For other data synthesis scenarios, especially load data training, the amount of time it takes for the losses to stabilise are usually much longer and the range of oscillation is wider. This is attributed to the higher variability of load power profiles compared to PV power and more parameters used in the DCGAN architecture for load synthesis.

Computational costs of the model fitting/training processes for the DCGAN, copula and constant normalised profile approaches are compared in Table 5.2.

Table 5.2: Computational costs for the training/model fitting process of the DCGAN, copula and constant normalised profile approaches regarding various data synthesis scenarios. The GPU memory usage is also shown for the DCGAN approach, whereas the other two methods can not use the GPU.

Unconditional Data Synthesis			
Data type	PV	Load	Net
DCGAN (hours)	2.33 (1497 MB)	33.79 (1497 MB)	10.08 (1497 MB)
copula (hours)	1.47	3.84	1.53
constant normalised profile (seconds)	14	13	15
Conditional Data Synthesis			
Data type	PV	Load	Net
DCGAN (hours)	5.64 (1497 MB)	31.95 (1497 MB)	29.56 (2489 MB)
copula (hours)	1.39	4.02	1.58
constant normalised profile (seconds)	9	9	11

The model fitting processes of the copula and constant normalised profile approaches are also performed on the same computer. The DCGAN takes a much longer time to train than the other two approaches, especially for training on the load power data. However, once the DCGAN model is trained, it takes negligible time for it to generate synthetic profiles. In fact, for all the data synthesis scenarios considered in this study, it takes no longer than 20 seconds to generate a single channel (PV/load/net) of synthetic profiles for all the households in the evaluation set. The copula and constant normalised profile approaches take less time to do a similar model inference, although the differences are relatively small among these three approaches.

5.3.2 Validation of synthetic data

Our main focus is to statistically evaluate the quality of the raw synthetic data generated by DCGAN and copula approaches against the real data in the evaluation set in this subsection.

Cumulative distribution functions (CDFs) are derived from the real and synthetic datasets to illustrate the distances between the measured and synthetic data probability distributions. The Jensen-Shannon divergence (JSD) [207] defined in Eqn. 5.9 is also calculated for the DCGAN and copula generated datasets, which measures their corresponding distances to the evaluation set probability distribution. Figure 5.9, Figure 5.10 and Figure 5.11 respectively show the cumulative distribution functions (CDFs) for PV, load and net meter

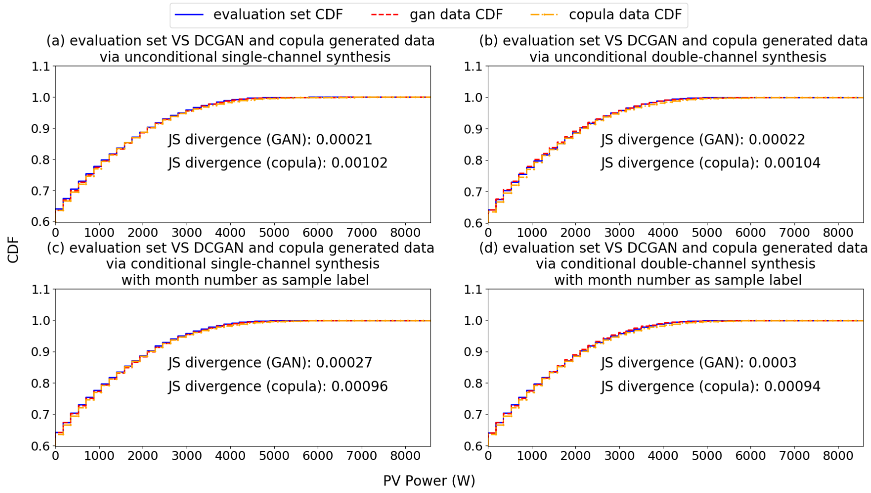

Figure 5.9: Cumulative distribution functions (CDFs) of the evaluation set PV power data, DCGAN and copula generated synthetic PV power data using (a) unconditional single-channel data synthesis, (b) unconditional double-channel data synthesis, (c) conditional single-channel data synthesis with month number as input label and (d) conditional double-channel data synthesis with month number as input label

power data, compared among the evaluation set, synthetic data generated by the DCGAN and copula approaches.

$$JSD(P||Q) = \sum_{x \epsilon X}[P(x)log(\frac{P(x)}{Z(x)}) + Q(x)log(\frac{Q(x)}{Z(x)})] \tag{5.9}$$

Where P and Q are the probability distributions of the measured and interpolated data defined on the same probability space X, $Z = \frac{1}{2}(P+Q)$, x represents a possible outcome from X.

The synthetic power data generated by DCGAN has a closer overall probability distribution to the evaluation set compared to the copula approach, regardless of the modelled data type or synthesis scenario. As demonstrated in the CDF plots with corresponding JSDs, both approaches perform reasonably well at producing similar PV power CDFs to the evaluation set whereas the DCGAN approach significantly outperforms the copula model for load and net meter power data synthesis. Similar results can also be observed for other data synthesis scenarios. There is not much difference between single-channel or double-channel synthesis for both approaches regarding the distances between the synthetic and measured probability distributions. For conditional power data synthesis, providing month number to the DCGAN model does not yield a smaller JSD for synthesising PV/load/net meter data however the week-day-focus label leads to a better JSD for load power data synthesis. On the other

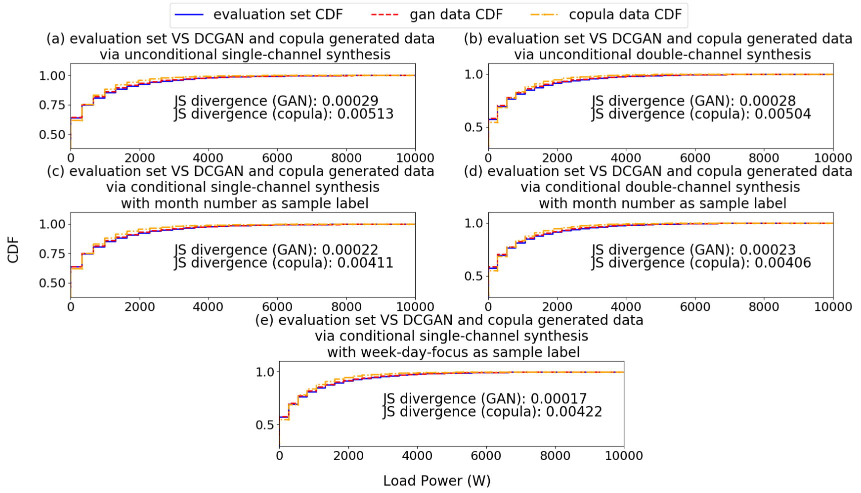

Figure 5.10: Cumulative distribution functions (CDFs) of the evaluation set load power data DCGAN and copula generated synthetic load power data using (a) unconditional single-channel data synthesis, (b) unconditional double-channel data synthesis, (c) conditional single-channel data synthesis with month number as input label, (d) conditional double-channel data synthesis with month number as input label and (e) conditional single-channel data synthesis with week-day-focus as input label

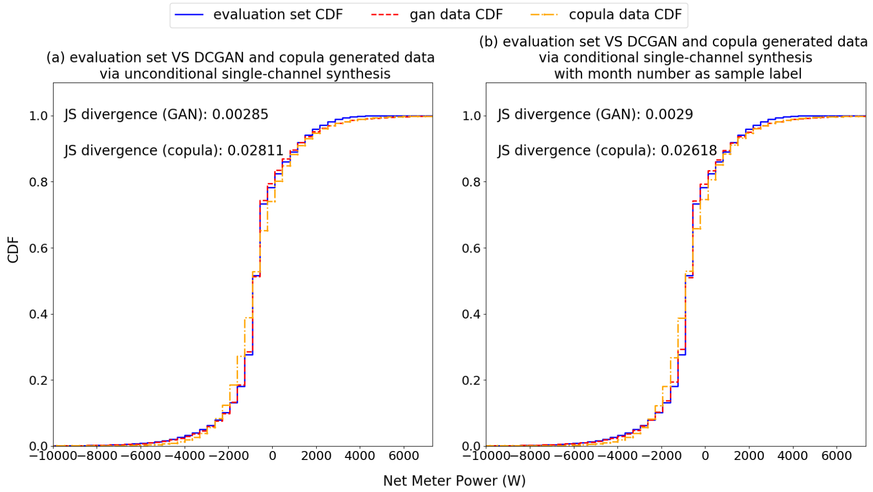

Figure 5.11: Cumulative distribution functions (CDFs) of the evaluation set net meter power data, DCGAN and copula generated synthetic Net meter power data using (a) unconditional single-channel data synthesis and (b) conditional single-channel data synthesis with month number as input label

hand, conditional data synthesis always leads to a better result for the copula method.

Mean daily autocorrelation profiles are computed to test how well the synthetic data captures the temporal characteristics of PV/load/net meter power data. To compute the mean autocorrelation daily profile, autocorrelations are derived from each 5-minute daily profile in the evaluation and synthetic datasets, then they are averaged for each 5-minute timestamp of a day. The mean daily autocorrelation profiles of the PV, load and net meter power data are respectively illustrated in Figure 5.12, Figure 5.13 and Figure 5.14 The DCGAN

for the measured data, synthetic data generated by the copula and DCGAN approaches under various data synthesis scenarios.

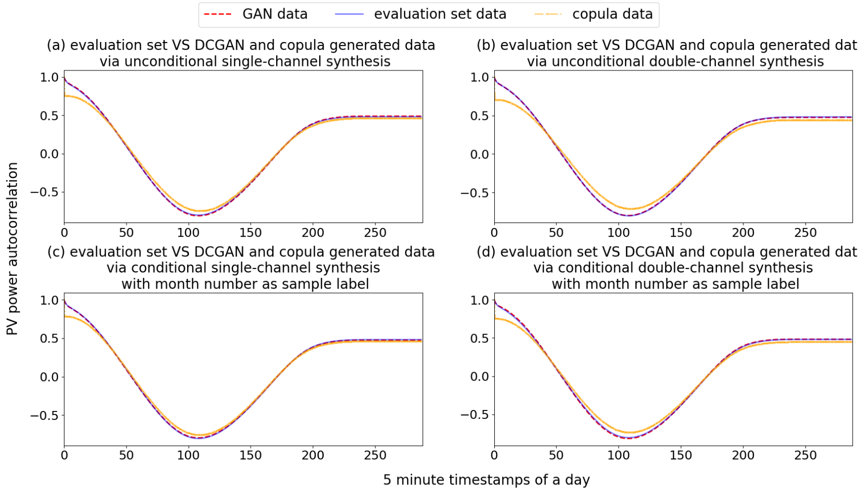

Figure 5.12: Mean daily autocorrelation profiles of the PV power data in the evaluation set, DCGAN and copula generated synthetic data using (a) unconditional single-channel data synthesis, (b) unconditional double-channel data synthesis, (c) conditional single-channel data synthesis with month number as input label and (d) conditional double-channel data synthesis with month number as input label.

generated PV power data results in an averaged autocorrelation curve that is quite well matched to the evaluation set. In contrast, there is some minor misalignment between the copula and the evaluation set's mean autocorrelation profiles. Although the DCGAN generated mean daily autocorrelation profiles of load and net meter power data also match the actual data autocorrelations fairly well, the autocorrelations for long time lags (>200 for net meter data, >250 for load data) tend to get underestimated. On the other hand, the copula derived mean autocorrelation plot of load power has a closer match for long time lags but not close for time lags that are less than 150. Furthermore, significant mismatch can be observed between the mean autocorrelation plots of the copula

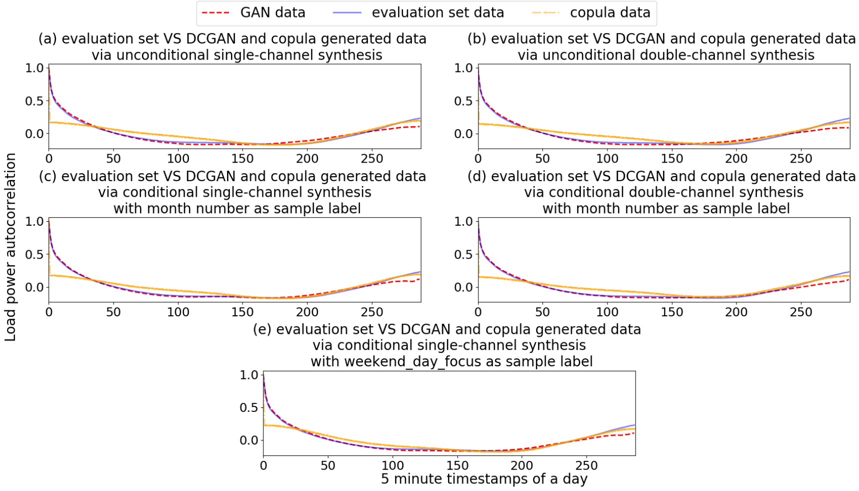

Figure 5.13: Mean daily autocorrelation profiles of the load power data in the evaluation set, DCGAN and copula generated synthetic data using (a) unconditional single-channel data synthesis, (b) unconditional double-channel data synthesis, (c) conditional single-channel data synthesis with month number as input label, (d) conditional double-channel data synthesis with month number as input label and (e) conditional double-channel data synthesis with week-day-focus as input label.

generated data and the evaluation set for net meter data synthesis scenarios.

Five-minute standard deviations are also calculated from the 5-minute average power values to evaluate whether the synthetic profiles present the same level of diversity as the measured profiles. As a result, Figure 5.15, Figure 5.16 and Figure 5.17 respectively demonstrate the 5-minute standard deviations of the PV, load and net meter power profiles for the ground truth and each evaluated synthesis scenario using the copula and DCGAN methods. In terms of the standard deviations in PV power data, both approaches capture the standard deviations fair well. The copula approach always overestimates the standard deviations whereas the DCGAN approach has some minor underestimations and overestimations. On the other hand, the copula model is clearly underestimating the load power standard deviations while the DCGAN approach has a much closer match for all the included load synthesis scenarios. For the standard deviations in net meter data, both approaches underestimate the standard deviations in the early morning and overestimating the standard deviations in the middle of a day. It is suspected this may be due to a different normalisation approach used for net meter power data. Either PV system size and peak load value are used for normalisation depends on which one is higher but they could lead to quite different normalised net meter profiles. Moreover, during the training process, this information on whether PV size or peak load is used for

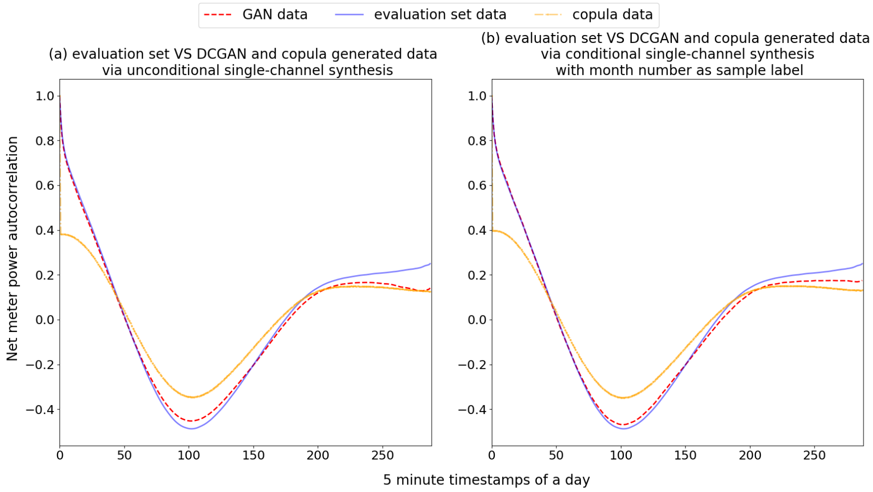

Figure 5.14: Mean daily autocorrelation profiles of the net meter power data in the evaluation set, DCGAN and copula generated synthetic data using (a) unconditional single-channel data synthesis and (b) conditional single-channel data synthesis with month number as input label.

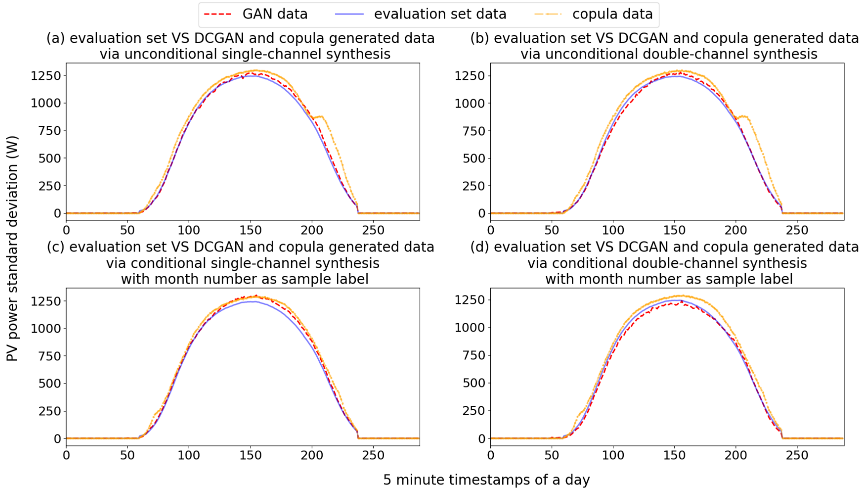

Figure 5.15: Standard deviations of the 5-minute PV power profiles in the evaluation set, DCGAN and copula generated synthetic data using (a) unconditional single-channel data synthesis, (b) unconditional double-channel data synthesis, (c) conditional single-channel data synthesis with month number as input label and (d) conditional double-channel data synthesis with month number as input label.

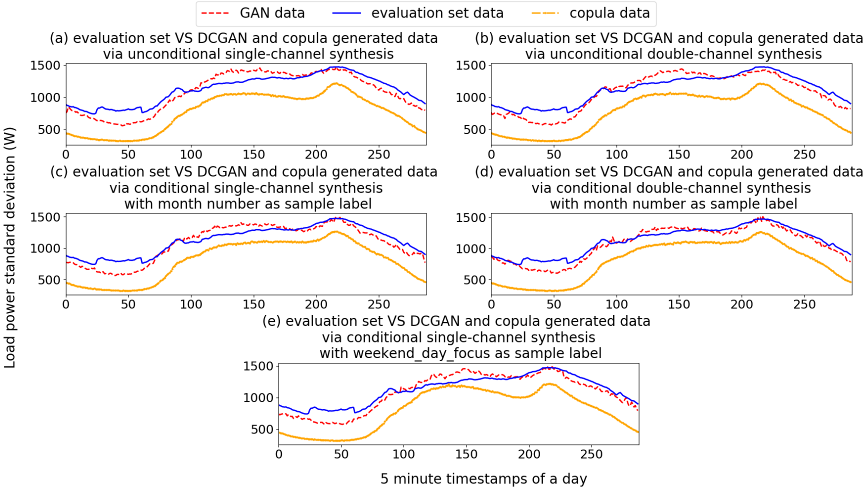

Figure 5.16: Standard deviations of the 5-minute load power profiles in the evaluation set, DCGAN and copula generated synthetic data using (a) unconditional single-channel data synthesis, (b) unconditional double-channel data synthesis, (c) conditional single-channel data synthesis with month number as input label, (d) conditional double-channel data synthesis with month number as input label and (e) conditional double-channel data synthesis with week-day-focus as input label.

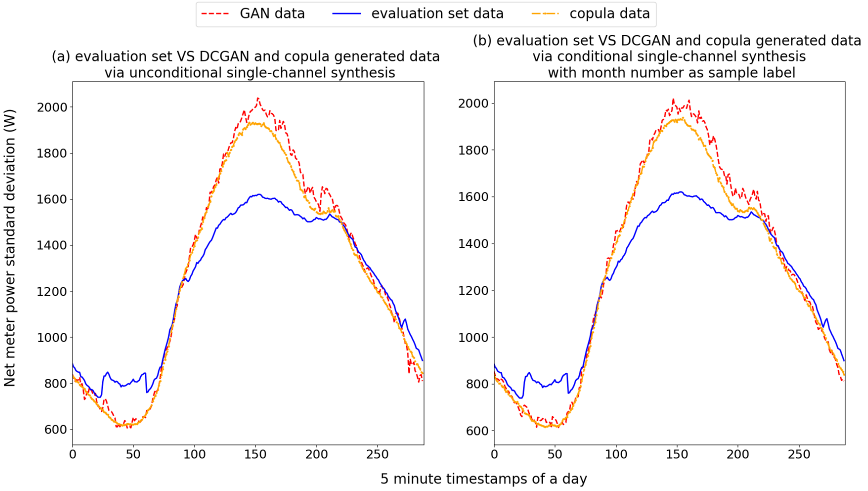

Figure 5.17: Standard deviations of the 5-minute net meter power profiles in the evaluation set, DCGAN and copula generated synthetic data using (a) unconditional single-channel data synthesis and (b) conditional single-channel data synthesis with month number as input label.

normalisation is never fed into the model. Hence, we believe this inconsistency may have caused some difficulties for the DCGAN and copula models to capture the targeted data distribution.

Normalised root mean squared errors (NRMSEs) in yearly totals are also used to evaluate how well the generative models could capture the aggregated sums for an individual household. After the synthetic data is generated for households in the evaluation set, the annual total PV/load/net meter energy values are derived by aggregating the power values for each household. Then the estimated yearly total energy values from the synthetic data are compared to the measured yearly total values for the households in the evaluation set to derive the normalised root mean squared errors (NRMSEs).

Table 5.3 shows the NRMSEs in annual total PV/load/net meter energy values for each evaluated data synthesis scenario. it is evident that the DCGAN approach outperforms the copula model for almost all the evaluated data synthesis scenarios except for the case of conditional net meter data synthesis. This means DCGAN performs better at estimating the aggregated energy sums at the individual household's level. Double-channel data synthesis without any input labels produces the closest estimations on yearly total energy for both DCGAN and copula approaches for PV data synthesis. In contrast, double-channel data synthesis with month number as the input labels produce the best results for load power data synthesis. Unconditional data synthesis leads to the lowest NRMSE in yearly net meter energy totals for the DCGAN method, which is the opposite to the copula results where conditional net meter data synthesis results in a minor error. Overall, PV data synthesis seems to produce the slightest error regarding yearly totals, whereas the net meter data synthesis has the highest NRMSEs.

Table 5.3: Normalised root mean squared errors (NRMSEs) in yearly total (a) PV, (b) net meter and (c) load energy when adopting synthetic power data generated by DCGAN and its percentage improvement in NRMSEs compared to using copula under various data synthesis scenarios. Positive improvement means the NRMSE by using DCGAN is smaller compared to using the copula model and the percentage is calculated by taking the difference and divide it by the NRMSE of using the copula approach. The smallest NRMSEs for PV/net meter/load power data synthesis are indicated by bold text.

(a) PV power data synthesis				
Data synthesis scenario	unconditional single-channel	unconditional double-channel	conditional single-channel (month number)	conditional double-channel (month number)

NRMSE in annual totals using GANs	0.058	**0.0552**	0.0567	0.0567
Improvement in NRMSE over copula	1.19%	4%	3.24%	3.08%
(b) net power data synthesis			**(c) load power data synthesis**	
Data synthesis scenario	unconditional single-channel	conditional single-channel (month number)	unconditional single-channel	unconditional double-channel
NRMSE in annual totals using GANs	**0.1335**	0.1358	0.1136	0.1139
Improvement in NRMSE over copula	1.18%	-1.12%	2.32%	2.9%
(c) load power data synthesis				
Data synthesis scenario	conditional single-channel (month number)	conditional single-channel (week-day-focus)	conditional double-channel (month number)	
NRMSE in annual totals using GANs	0.1125	0.1152	**0.1124**	
Improvement in NRMSE over copula	3.43%	5.5%	2.26%	

To summarise, the proposed DCGAN approach can perform well on these metrics and at the same time outperforms the comparative copula approach for almost every included data synthesis scenario. This provides sufficient evidence that the DCGAN model can adequately model the targeted PV/load/net meter power data distributions and is also able to generate realistic samples (shown in Figure 5.5).

As the proposed DCGAN model can generate high-quality samples with no statistical requirements and minor requirements of input information, it can be potentially integrated into some network planning/optimisation studies which require the sub-station levels of load/PV power curves aggregated from a large number of individual household load/generation profiles. Moreover, it could potentially be used to generate a large amount of realistic PV/load power scenarios for reinforcement learning based microgrid/battery storage power scheduling optimisation studies which requires sufficient amount of power scenarios to allow the software agents to learn how to make scheduling decisions towards the optimisation goals.

However, in terms of the comparisons between various data synthesis scenarios for PV/load/net meter data, it is still unclear that whether using single-channel or double-channel data synthesis is better or whether conditional is superior to unconditional data synthesis as none of these synthesis scenarios is consistently outperforming the others across the above metrics. Moreover, even though the DCGAN generated data bears more statistical resemblance to the real data compare to the data generated by the copula approach, it is also important to test these models in an end-us application and compare their performances. Hence, it is vital to use more concrete metrics designed for the intended end-use applications such as a battery sizing model to further assess the two approaches under various data synthesis scenarios and to quantitatively recommend.

5.3.3 End-use application validation

The end-use application for the synthetic PV and load power data in this study is to estimate electricity costs and to size battery storage systems for new solar households with no historical data. Furthermore, we consider the cases where a customer may have access to either measured PV or load interval data. As mentioned in Section 5.2.4, we also evaluate another comparative case where normalised profiles are used for the battery sizing model, a common practice in the industry.

Errors in electricity costs & battery savings

Figure 5.18 shows the NRMSEs in annual electricity costs when using synthetic PV & load power data generated by DCGAN, copula models and constant normalised profiles via various data synthesis scenarios. In these plots, we categorise the synthetic PV & load power data synthesis cases into three main groups: (1) unconditional data synthesis (single & double channel synthesis); (2) conditional data synthesis with month number as input label for both load & PV (single & double channel synthesis); (3) conditional single-channel data synthesis with month number as input label for PV data synthesis and week-day-focus as labels for load profiles.

The DCGAN approach outperforms the other two models by small margins for each evaluated data synthesis scenario under both ToU and flat tariff settings. The major difference between the DCGAN approach and constant normalised profile approach is that the latter approach takes the mean profile of the training set, as opposed to the DCGAN approach which generates random samples from the learned training data distribution. Therefore, the use of averaged normalised profiles is expected to end up with low and stable overall error compared to different measured customer profiles in the evaluation set that has similar means to the training set. On the other hand, the DCGAN generated profiles are more diverse which could potentially lead to higher overall mean squared errors in costs. Hence, it shows that the DCGAN approach captures targeted PV & load data distributions well under various data synthesis scenarios.

Conditional data synthesis ends up with smaller NRMSEs for DCGAN compared to unconditional data synthesis where conditional double-channel synthesis seems to produce the best results, although we discover that these data synthesis scenarios do not make too much differences in terms of the errors in estimating electricity costs for new customers.

The NRMSEs in yearly electricity costs for using synthetic PV + measured load and measured PV + synthetic load power data are respectively displayed in Figure 5.19 and Figure 5.20. For applying synthetic PV & real load data, DCGAN still results in the lowest errors in electricity costs under both tariffs. Conditional data synthesis always seems to be the better option in generating synthetic PV power data and this holds for all three tested approaches. As shown in Figure 5.20, The DCGAN and constant normalised profile approaches both generate smaller errors in electricity costs compared to the copula approach when using real PV & synthetic load power data. There is minimal difference between the results of DCGAN and constant normalised profile models where for some cases the latter approach outperforms the DCGAN by a tiny margin, this is somehow expected for using an averaged profile as explained above. Conditional data synthesis with month number always end up with a smaller NRMSE compared to unconditional synthesis for DCGAN using synthetic load & real PV data. In contrast, there is no obvious benefits of using conditional data synthesis for the copula and constant normalised profile approaches.

We calculate battery savings by taking the differences between the electricity costs before and after battery storage systems are installed, as a result, Figure 5.21 demonstrates the NRMSEs in yearly battery savings using synthetic PV & load power data generated by the DCGAN, copula and constant normalised profile approaches under various data synthesis scenarios. It is clear that the DCGAN approach leads to much less errors in battery savings compared to the other two models for both tariff structures. This is expected as the previous results have shown DCGAN is able to generate closer samples compared to the copula approach. Moreover, different to just apply averaged constant profiles, DCGAN can model the intrinsic variability of residential PV/load profiles. Hence for an individual household, the constant normalised prole model ends with the same import/export prole for each day of a year. In contrast, the DCGAN approach leads to highly stochastic import/export proles. This does not make too much of a difference when calculating electricity costs as both models capture the total import/export energy well: DCGAN generated data produces very similar load & PV power CDFs as shown in the previous subsection; the constant normalised profile model is using the mean profile from the target power data distribution. However, when calculating battery savings, using a constant mean profile is not desirable as it could produce misleading results on how much battery capacity is utilised. For instance, a household could have a week of high generation & low consumption and the following week with low generation & high consumption. By just taking a mean import/export pro-

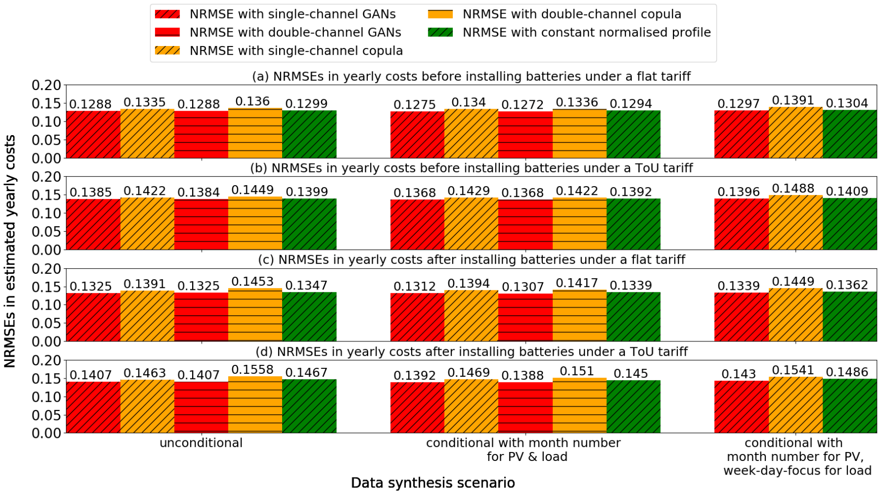

Figure 5.18: NRMSEs in yearly electricity costs before installing batteries under (a) a flat tariff or (b) a ToU tariff and after installing batteries under (c) a flat tariff or (d) a ToU tariff using synthetic PV & synthetic load power data generated via various data synthesis scenarios.

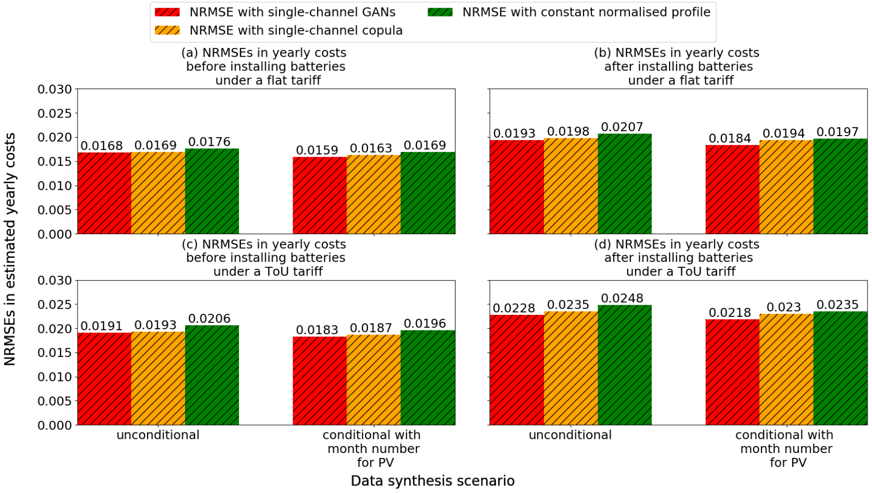

Figure 5.19: NRMSEs in yearly electricity costs before installing batteries under (a) a flat tariff or (c) a ToU tariff and after installing batteries under (b) a flat tariff or (d) a ToU tariff using synthetic PV power data via various PV data synthesis scenarios & real load power data.

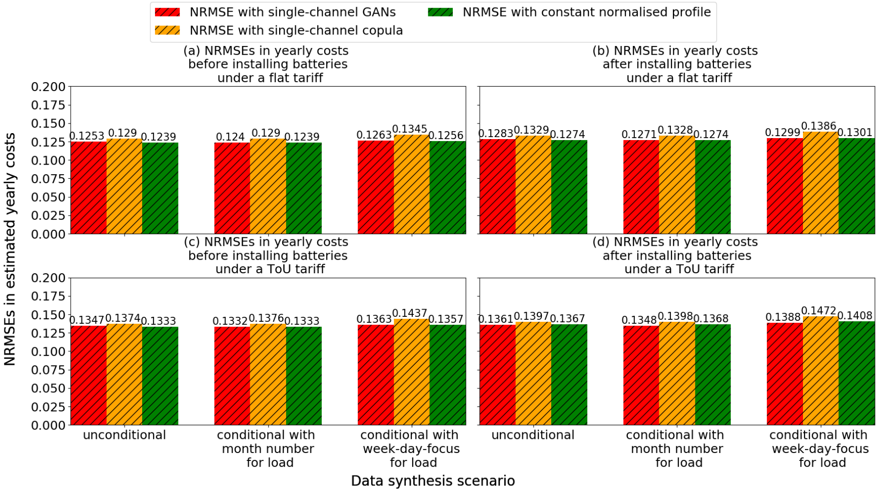

Figure 5.20: NRMSEs in yearly electricity costs before installing batteries under (a) a flat tariff or (c) a ToU tariff and after installing batteries under (b) a flat tariff or (d) a ToU tariff using real PV & synthetic load power data generated via various load data synthesis scenarios.

file, it is assumed that for each day, a certain amount of battery capacity can be used to increase the self-consumption of PV generation, thus saves on electricity costs. On the other hand, the real situation might be that the battery will be mostly full in the first week and mostly empty for the following week so that only a small amount of battery capacity is actually utilised. By producing import/export daily profiles with the right amount of diversity, the DCGAN approach potentially produces more realistic simulated battery operation profiles, which lead to lower errors in battery savings.

Feeding in month number and week-day-focus labels respectively for DCGAN training of PV and load power data seems to have the most negligible errors in savings. It also leads to the lowest NRMSE for the other two approaches.

Figure 5.22 and Figure 5.23 respectively illustrate the NRMSEs in electricity savings using synthetic PV + real load and real PV + synthetic load power data generated by the DCGAN, copula and constant normalised profile approaches for various data synthesis scenarios. For these two circumstances, DCGAN also clearly outperforms the comparative models, regardless of the data synthesis scenario. Under both tariff structures, conditional DCGAN using month number for PV power data synthesis results in the smallest NRMSE in savings for households that only require synthetic PV data for battery sizing. For synthetic load + real PV data, conditional DCGAN using the input label of week-day-focus leads to the lowest NRMSE.

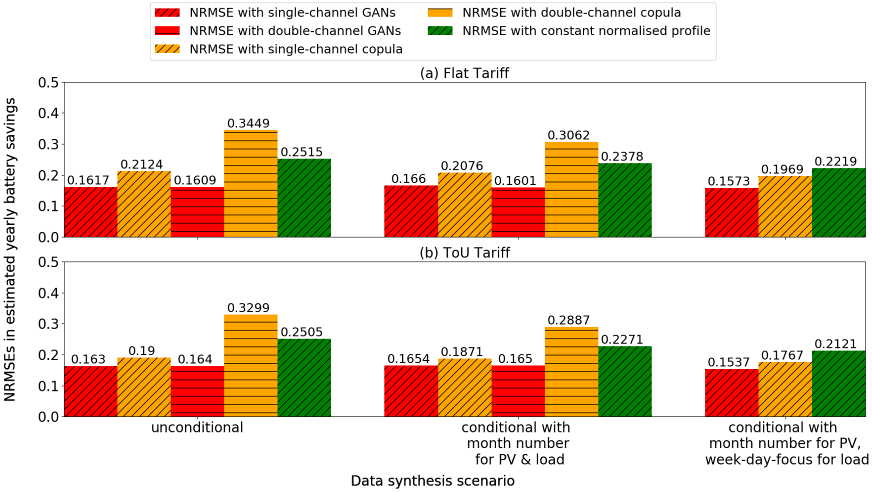

Figure 5.21: Under (a) a flat tariff or (b) a ToU tariff, NRMSEs in yearly battery savings using synthetic PV & synthetic load power data generated via various data synthesis scenarios.

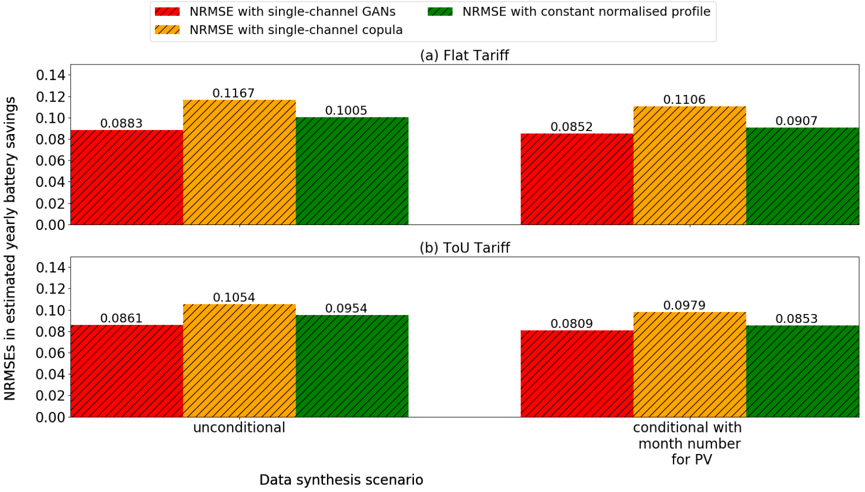

Figure 5.22: Under (a) a flat tariff or (b) a ToU tariff, NRMSEs in yearly battery savings using measured load power data & synthetic PV power data generated via various synthesis scenarios.

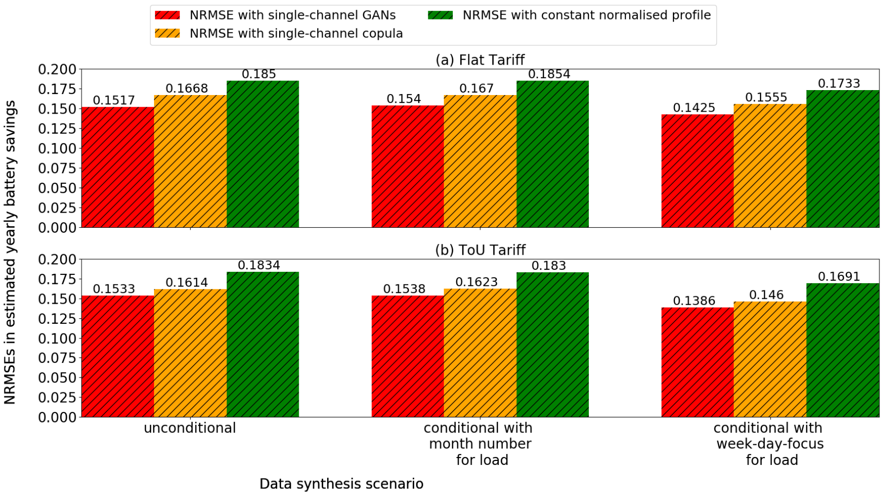

Figure 5.23: Under (a) a flat tariff or (b) a ToU tariff, NRMSEs in yearly battery savings using measured PV power data & synthetic load power data generated via various data synthesis scenarios.

Errors in optimal battery sizes

As mentioned earlier, to select the optimal battery size, a grid search is performed to select the battery size that leads to the highest NPV at the end of the battery lifetime. Figure 5.24 shows the mean absolute errors (MAEs) in estimated optimal battery sizes when using synthetic PV & synthetic load data generated by the DCGAN, copula and constant normalised profile approaches under various data synthesis scenarios. Similar to the results for costs and savings, DCGAN results in the smallest MAEs in optimal battery sizes for both tariff settings and various data synthesis scenarios. The constant normalised profile approach produces much higher errors compared to the copula and the DCGAN models, we suspect the reason behind it is the issue mentioned earlier which is the misleading simulated battery operation profiles generated by using a constant profile for each day instead of diverse profiles produced by randomly sampling from the targeted power data distributions. Conditional DCGAN with month number as sample label for PV profile and week-day-focus for load profile produces the lowest errors for both tariff structures, this is expected as this synthesis setting also generates the smallest errors in battery savings. For the situation where there is no week-day-focus label available, the best option seems to be conditional double-channel DCGAN using month number as the power profile label.

Figure 5.25 and Figure 5.26 illustrate the mean absolute errors (MAEs) in estimated optimal battery sizes when using synthetic PV + real load and real PV + synthetic load power data under various data synthesis scenarios. The

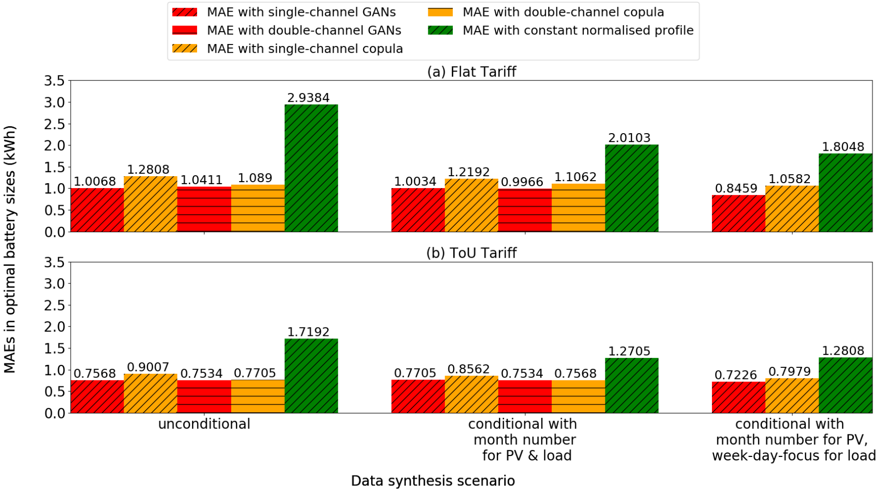

Figure 5.24: Under (a) a flat tariff or (b) a ToU tariff, MAEs in estimated optimal battery sizes using synthetic PV & synthetic load power data generated via various data synthesis scenarios.

DCGAN approach still leads to the smallest MAEs, similar to the results on battery savings errors. Figure 5.26 indicates that week-day-focus is a better input label for synthesising load profiles compared to month number. As shown in Figure 5.25, conditional synthesis on PV power data leads to better sizing results under the ToU tariff but worse errors for the flat tariff. Hence, we can conclude that providing month number does not help too much for single-channel PV data synthesis, at least for battery sizing.

The above results also provide recommendations over various data synthesis scenarios: conditional DCGAN conditioned on month number for PV data synthesis and week-day-focus for load data synthesis results in the best battery sizing results followed by conditional double-channel DCGAN conditioned on month number which requires less information from new customers. Moreover, for conditional DCGAN, double-channel architecture produces better results compared to single-channel probably as it can take advantage of the correlation between load and PV profiles.

The effectiveness of the DCGAN approach on estimating electricity costs and sizing home energy storage systems shows its potential of being a more accurate yet simple battery sizing tool for residential customers who are making battery purchase decisions or battery system installers/electricity retailers who often provide quoting services for new customers.

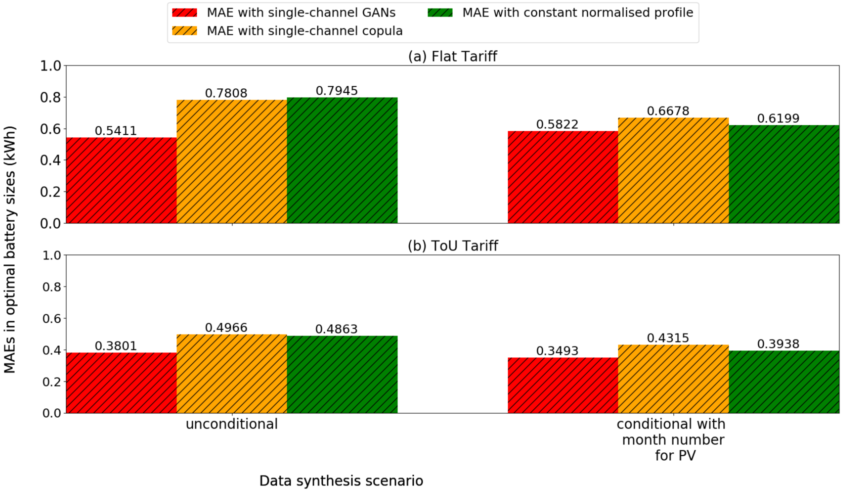

Figure 5.25: Under (a) a flat tariff or (b) a ToU tariff, MAEs in estimated optimal battery sizes using synthetic PV power data generated via various PV data synthesis scenarios & measured load power data.

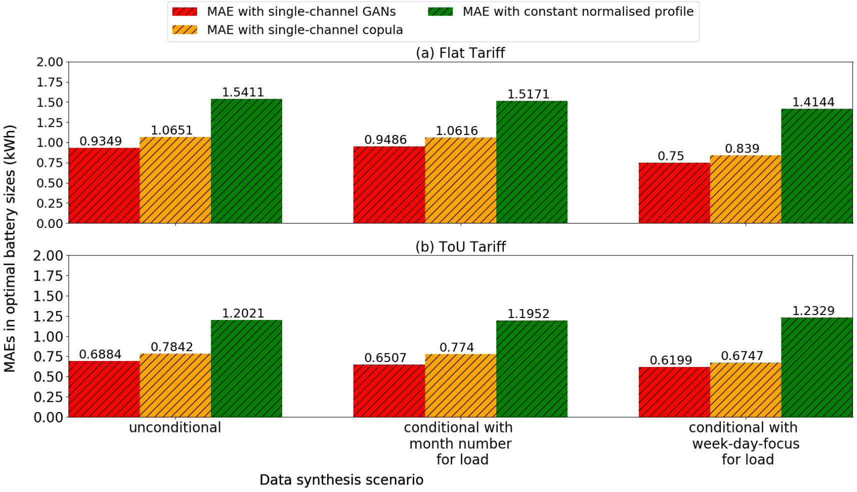

Figure 5.26: Under (a) a flat tariff or (b) a ToU tariff, MAEs in estimated optimal battery sizes using measured PV power data & synthetic load power data generated via various data synthesis scenarios.

5.4 Summary

In this chapter, we propose a DCGAN based model, which aims to generate synthetic PV/load/net meter power profiles for residential customers without any historical data. Two levels of validations are performed for the synthetic data generated by the DCGAN model using the evaluation dataset of 292 households: statistical evaluations to test how well the DCGAN model can generate samples from the desired data distribution, a comparative copula approach is also evaluated; evaluate the performance of DCGAN on an end-use application which is to estimate electricity costs and size battery systems for new PV households with no historical data, the results are compared with the copula approach and the constant normalised profile model which is a common industrial practice.

As demonstrated in figures in Section 5.3.2, the developed model can generate highly realistic 5-minute synthetic power profiles that are statistically akin to real profiles. Furthermore, the results shown in Figure 5.18 and Figure 5.21 suggest that the DCGAN generated synthetic dataset can be applied in a real time end-use application and surpasses the performances of the models used in the relevant literature and the industry. Moreover, it is also demonstrated in Figure 5.19, Figure 5.20, Figure 5.22 and Figure 5.23 that when one channel of synthetic data is accessible (PV or load), the DCGAN method still outperforms the alternative models with regards to battery sizing results. The proposed model can be integrated to other research that requires a large amount of power scenarios or to a battery sizing tool that only requires PV system size and peak electricity power as inputs for residential households, installers and utilities.

Chapter 6

Interpolating High-granularity PV and Load Data using Super Resolution Generative Adversarial Networks

6.1 Introduction

Solar generation and load consumption data, especially in the residential sector, is highly stochastic due to system location, local weather, socioeconomic factors and occupant behaviours. The large amount of high-resolution data collected by smart meters could be used to model and forecast PV generation and load consumption. However, the downsides of collecting high-resolution data are the additional costs of storing, transferring and managing the collected datasets and privacy concerns. As a result, the common temporal resolutions of smart meter data are still 15-minute or coarser [11]. This level of temporal resolution might be sufficient for billing purposes or deriving the overall generation/consumption pattern. It can not fully capture the generation and consumption spikes as illustrated in power profiles in Figure 6.1. Furthermore, this may lead to inaccuracies in the modelling and optimisation of distributed generation systems.

The impacts of adopting coarse datasets on distributed generation system optimisation have been investigated in several studies reviewed in Chapter 2. The analysis done in Chapter 3 also showed that under a flat tariff, the resulting discrepancies between 5-second data and hourly data are 2.9% in estimated electricity costs and 12.6% in battery savings on average. 5-minute resolution is recommended which achieves a satisfactory balance between accuracies and

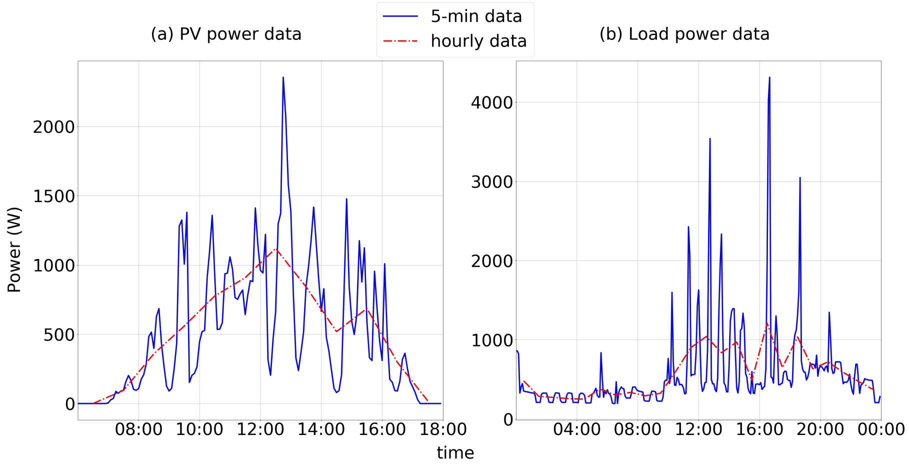

Figure 6.1: comparisons between 5-minute and hourly resolutions for (a) a PV power profile on a cloudy day and (b) a residential load power profile.

computational costs.

To address the above issues caused by low granularity data, one potential approach is to interpolate high-resolution load and PV data from lower resolution data. However, there is a quite limited amount of literature regarding this topic. Regarding PV power interpolation, the most relevant studies attempt to interpolate high-resolution solar irradiance data from low-resolution measurements. In [208], a model is proposed to generate 10-minute irradiance data from hourly measurements where the stochastic component of the 10-minute data is reproduced by randomly generating fluctuations from fitted beta distributions. Some improvements have been made in [209] and [210], reducing the errors between the synthetic and measured high-resolution solar irradiance data. These studies all require an indicator (normalised clearness index in [208], beam clearness index in [209] and clear sky index in [210]) based on irradiance measurements to classify sky conditions. Hence, as irradiance measurements are difficult to obtain especially for residential sites, the practicability of these approaches is questionable for interpolating PV power data. Regarding load data, the nearest related study sought to improve load disaggregation accuracy by interpolating high frequency load data (100/1000 Hz) from lower frequency load data (10/100 Hz) using a convolutional neural network (CNN) trained by mean squared error (MSE) [211].

Overall, to the authors' knowledge, there are no existing studies that interpolate high-resolution PV/load power data from coarse smart meter data that is commonly accessible in practice (e.g. 30-minute/hourly). In this chapter, inspired by the super resolution generative adversarial network (SRGAN) work proposed in [212] which sets a new state-of-art for image super-resolution, a deep learning model is proposed to interpolate 5-minute PV generation and load consumption power data from 30-minute/hourly smart meter measurements. The interpolated data is then adopted in a residential PV battery optimisation model to address the inaccuracies in the optimised results caused by using coarse data. The reasons for setting the targeted temporal resolution to 5-minute are two-folds: 1. 5-minute is sufficient for applications investigated in [11, 50, 114] and is recommended in Chapter 3, which achieves a good balance between accuracy and computational costs for optimisation of a PV battery system; 2. Although the model can be easily adjusted to generate higher resolution data, the amount of higher resolution data required to fit the model is also larger and may not be easily accessible in practice.

The source code for the implementation, together with the trained parameters of the proposed SRGAN model are available online at `https://github.com/tomtrac/SRGAN\_power\_data\_generation`. This allows others to: easily apply our SRGAN model to their own datasets; apply our trained network directly to generate 5-minute data from 30-minute/hourly measurements; and to compare their results with this work.

The remainder of the chapter is organised as follows: Section 6.2 illustrates the problem formulation of the data interpolation; Section 6.3 introduces the proposed model; Section 6.4 presents detailed evaluations of the interpolated data; Section 6.5 concludes the study and proposes some future work.

6.2 Problem Formulation

The interpolation aims to estimate a high temporal resolution average power generation/consumption profile X^{HR} from its lower resolution version X^{LR}. X^{LR} is essentially a time series with M average power values and with an upsampling factor u which means u power values are interpolated from a single value in X^{LR}, as a result X^{HR} contains $u \times M$ time-indexed values.

To set up the interpolation model, historical high-resolution data collected from multiple sites are used in the training set to train a generating function G_{θ_G} parameterised by θ_G. The training task can be defined as finding θ_G^* in Eqn. 6.1:

$$\theta_G^* = \arg\min_{\theta_G} \frac{1}{N_{train}} \sum_{n=1}^{N_{train}} J_G(G_{\theta_G}(X_n^{LR}), X_n^{HR}) \tag{6.1}$$

Where θ_G^* are the optimal parameters that minimise G_{θ_G}'s loss function J_G described in details in Section 6.3.2; X_n^{HR} and X_n^{LR} respectively denote a single high-resolution and a low-resolution PV generation/load power profile in the training set; N_{train} is the total number of power profiles in the training set and $n = 1, ..., N_{train}$.

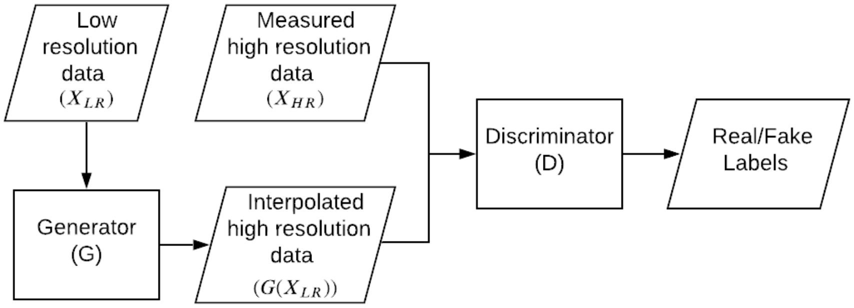

Figure 6.2: The structure of the SRGAN.

6.3 Methodology

Generative adversarial networks (GAN), as a machine learning framework introduced in Chapter 5, forms the basis of SRGAN. In this section, the concept of the SRGAN is briefly introduced, then the loss function, architecture and training process of the proposed SRGAN model are described.

6.3.1 Super Resolution Generative Adversarial Networks

For the interpolation task considered in this study, the structure of the original GAN is adjusted to a SRGAN shown in Figure 6.2: instead of latent noises, low-resolution power profiles are inputted to the generator to generate high-resolution power profiles; then the discriminator's task is to distinguish synthetically interpolated power profiles from real high-resolution power profiles.

6.3.2 Loss Function

During the model training process, the discriminator aims to maximise the probability of assigning the correct labels to measured high-resolution power profiles (X^{HR}) and interpolated profiles ($G(X^{LR})$). This is done by minimising the cross-entropy cost $J_D(\theta_D, \theta_G)$ shown in Eqn. 6.2, given that θ_D, θ_G are respectively the parameters of the discriminator and generator:

$$J_D(\theta_D, \theta_G) = -\mathbb{E}_{X^{HR} \sim p_{HR}}[\log D(X^{HR})] - \mathbb{E}_{X^{LR} \sim p_{LR}}[\log(1 - D(G(X^{LR})))] \tag{6.2}$$

Where p_{HR} and p_{LR} represent the data distributions of the high and low-resolution power profiles respectively.

The loss function of the generator, on the other hand, has two main components: one is the MSE between the interpolated and measured high-resolution data (shown in Eqn. 6.3) which shall be minimised to make sure the reconstructed power values are close to the measured high-resolution values; the other loss is the adversarial loss (J_A shown in Eqn. 6.4) which is minimised during training to maximise the probability that the discriminator being mistaken. The reasons to include the adversarial losses are two-fold: 1. minimising just the MSE encourages finding the averages of the plausible interpolation solutions and this creates overly-smooth interpolation results that are not realistic [212], this issue is found in the area of image super-resolution; 2. adding adversarial loss encourages the generator to capture high-resolution uncertainties to make the interpolated profiles realistic enough to fool the discriminator.

$$J_{MSE}(\theta_G) = \frac{1}{uM}\sum_{t=1}^{uM}(X_t^{HR} - G(X^{LR})_t)^2 \tag{6.3}$$

Where J_{MSE} is the MSE loss, t represents a timestamp, X_t^{HR} and $G(X^{LR})_t$ are the corresponding power values in the measured and interpolated high-resolution power profile respectively.

$$J_A(\theta_D, \theta_G) = -\mathbb{E}_{X^{LR}\sim p_{LR}}[\log(D(G(X^{LR})))] \tag{6.4}$$

The combined loss J_G of the generator is the weighted sum of J_{MSE} and J_A:

$$J_G = J_{MSE} + \lambda \times J_A \tag{6.5}$$

Where λ is the weighting factor applied for the adversarial loss. As J_G and J_D are inversely correlated, they can combine and form a min-max objective V for both functions:

$$\begin{aligned}\min_G \max_D V(D,G) = \mathbb{E}_{X^{HR}\sim p_{HR}}[\log D(X^{HR})] + \mathbb{E}_{X^{LR}\sim p_{LR}}[\log(1 - D(G(X^{LR})))] + \\ \lambda \times (\frac{1}{uM}\sum_{t=1}^{uM}(X_t^{HR} - G(X^{LR})_t)^2)\end{aligned} \tag{6.6}$$

6.3.3 Model Architecture

The design of the proposed model architecture is inspired by the original SRGAN work in [212], where both the generator and discriminator are implemented as deep convolutional neural network (CNN). By applying multiple filters and stacked convolutional layers, hierarchical levels of temporal dependencies/features can be captured from the input image/time series without requiring extra hand-crafting preprocessing steps other than normalisation of the input data. As a result, deep CNNs have achieved many breakthroughs in the domains of image recognition [187] and restoration [213], speech recognition [214] and natural language processing [215].

However, as the depth of a CNN keeps increasing to a certain extent, often the model accuracy gets saturated and decreases rapidly. This degradation in model performance is addressed by residual neural network (ResNet) proposed in [216], which includes residual blocks that add skip connections along with the normal data flows in a deep CNN. Hence, in this study, residual blocks are applied for the generator to allow extra useful information to flow from the input data and at the same time avoid the degradation issue of very deep CNNs. The architecture of the adopted residual block is shown in Figure 6.3(a), which follows the work proposed in [212]. Each residual block includes two convolutional layers followed by batch normalisation, which standardises the outputs of the previous layers in order to stabilise and accelerate the training process [190]. Parametric rectified linear unit (PReLU) [217] is used as the activation function.

Consider the residual block input as x and the desired output of the residual block is $H(x)$. As shown in Figure 6.3(a) where a skip/identity connection is added to a stacked CNN, the input x is copied and added to the output of the stacked layers. This means instead of fitting these in-between layers directly to produce $H(x)$, another mapping $F(x)$ called a residual mapping is used where $F(x) = H(x) - x$. Hence $H(x)$ is recast into $F(x) + x$. The skip connections allow information to flow between layers easily without any transformations and help the later layers to utilise the information from the original input layer or previous layers. Moreover, the skip connections enable identity mappings (the output is the same as the input), which is difficult to approximate for traditional non-linear deep CNNs. Hence, if the optimal layer mapping is close to an identity mapping, this skip structure makes it easier to find the optimal layer parameters. To increase the resolution of the input data, the upsampling block shown in Figure 6.3(b) is applied, which is inspired by the work in [189] and includes an initial nearest-neighbour interpolation, a convolutional layer with batch normalisation and a rectified linear unit (ReLU) activation function [191].

The model architecture of the generator is shown in Figure 6.4(a), which includes M residual blocks and N upsampling blocks. A sigmoid activation layer is applied at the end to make sure the output normalised numerical range is between 0 and 1.

For the discriminator, as less convolutional layers are adopted, no residual blocks are required. The design shown in Figure 6.4(b) simply follows the guidelines proposed in [156] for a deep convolutional generative neural networks (DCGAN). Leaky ReLU [192] is applied as the activation function and batch normalisation is also applied.

6.3.4 Model Training

Both the generator and the discriminator are trained by backpropagation [218] with multiple training iterations. For each iteration, the following steps are performed to update the parameters of both functions:

1. A mini-batch that consists of multiple LR and HR daily PV/load profile pairs is randomly drawn from the training data.

2. The generator parameters are kept constant. The mini-batch is used to update the discriminator parameters through backpropagating the loss

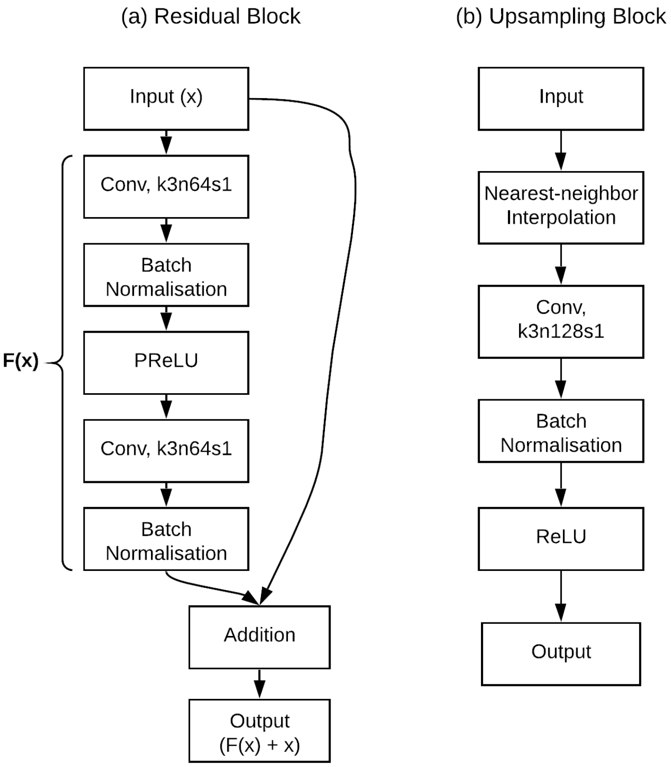

Figure 6.3: The structure of the adopted (a) residual block and (b) upsampling block. "Conv" refers to a convolutional layer, the numbers after "k", "n" and "s" respectively stand for the filter size, number of filters and stride amount of the convolutional layer(e.g. k3n64s1 indicates that the convolutional layer has a filter size of 3 × 3, 64 filters and a stride of 1.)

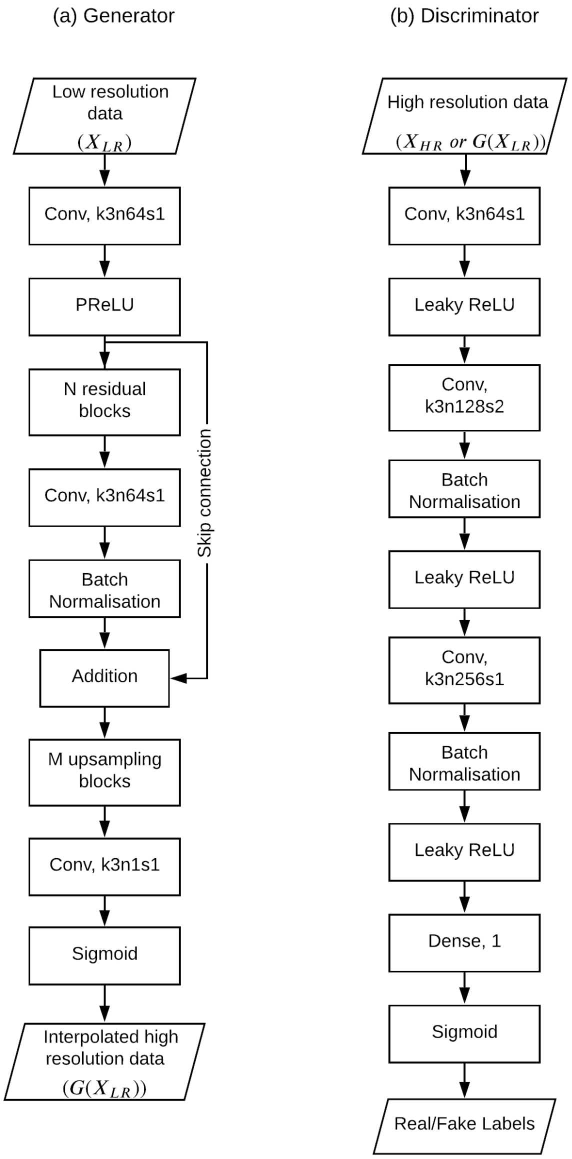

Figure 6.4: The architecture of the (a) generator and (b) discriminator. "Conv" refers to a convolutional layer, the numbers after "k", "n" and "s" respectively stand for the filter size, number of filters and stride amount of the convolutional (e.g. k3n64s1 indicates that the convolutional layer has a filter size of 3×3, 64 filters and a stride of 1.)

defined in Eqn. 6.2.

3. Another mini-batch is sampled from the training data.

4. The discriminator parameters are kept constant. The second mini-batch is used to update the generator parameters through backpropagating the loss defined in Eqn. 6.5.

6.4 Case Study

6.4.1 Dataset and Model Training

The dataset used in the case study includes 5-minute average PV generation and load consumption data of 2925 Australian PV households, collected by Solar Analytics [160] using Wattwatcher smart meters [161] for the period between January 2017 and December 2017. 5-minute data is then resampled into 30-minute and hourly datasets. The PV and load power data are normalised by the household's PV system size and peak load respectively before fitting the SRGAN model, this makes sure the numerical range is between 0 and 1. The power data from 80% of the households is used to train the SRGAN model, 10% as the test set to evaluate the performance of the model and the remaining 10% is used as the validation set to select the optimal model hyperparameters such as the number of training iterations, numbers of residual blocks and upsampling blocks in the generator.

Model training is performed on a PC with an Nvidia GeForce RTX 2070 GPU, an Intel Core i7-8700K CPU and 32 GB of RAM, using Keras [206] and Tensorflow [205] as the deep learning packages. The mini-batch size is set to be 128 and the optimiser to update the SRGAN's parameters is set to be Adam [193] with a learning rate of 10^{-4} and momentum β_1 of 0.5. Separate models are trained for interpolating 30-minute and hourly PV and load data, then some tuning of the model hyperparameters is done using the validation set and the Jensen-Shannon divergence (JSD) [207] as the evaluation metric, which measures the distances between the interpolated and the measured data probability distributions. Empirically it is found that 5 residual blocks are sufficient for all the evaluated interpolation scenarios. On the other hand, interpolating hourly PV/load data requires two upsampling blocks, while one upsampling block is sufficient for interpolating 30-minute data. The range for the optimal numbers of training iterations is between 10^5 to 5×10^5. More iterations are required for interpolating hourly data as it has an additional upsampling block. The weighting factor λ is set to be 10^{-3} in Eqn. 6.5.

6.4.2 Results and Discussion

Visual inspection

The first step of the model evaluation is to visually inspect the SRGAN generated profiles and their ground truth. Moreover, another aspect to evaluate

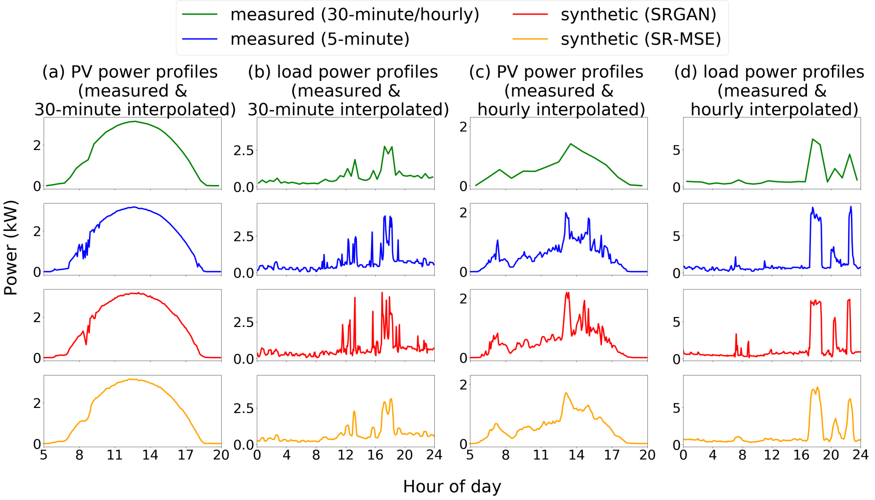

Figure 6.5: 5-minute daily PV power measured profiles compared to synthetic profiles generated by the SRGAN and SR-MSE models for (a) a clear-sky day interpolated from 30-minute data and (c) a cloudy day interpolated from hourly data; 5-minute daily load measured profiles compared to synthetic profiles interpolated from (b) a 30-minute power profile and (d) an hourly power profile. The input measured 30-minute/hourly profiles are shown in the first row.

is whether it is necessary to include the adversarial loss component in the loss function in Eqn. 6.5 for interpolating PV/load power data. Hence, in addition to the SRGAN model, another approach with the same model architecture is trained only using the MSE loss component in Eqn. 6.5 and this model is referred as the super resolution mean squared error (SR-MSE) approach.

As a result, Figure 6.5 shows a few examples of SRGAN and SR-MSE interpolated 5-minute PV and load power profiles and their respective input 30-minute/hourly and 5-minute measured power profiles. For an example of a clear-sky day PV power profile, as shown in Figure 6.5(a), the synthetic profile generated by SRGAN matches well with the measured profile. A cloudy day PV power profile is illustrated in Figure 6.5(b), although some discrepancies can be observed where the SRGAN interpolated profile does not match the measured profile point by point, it captures the overall pattern and variations in power quite well. Similar results can be observed from Figure 6.5(c) and 6.5(d), which compare two 5-minute load profiles respectively interpolated from 30 minutes and hourly resolutions using SRGAN, to their corresponding measured profiles. On the other hand, although the SR-MSE approach can capture the overall patterns of the PV and load profiles, its generated profiles seem to be too smooth and less convincing especially for load profiles and cloudy-day PV profiles. This shows the necessity of applying the adversarial loss for interpolating PV and load power data, similar to what is found for image super-resolution [212].

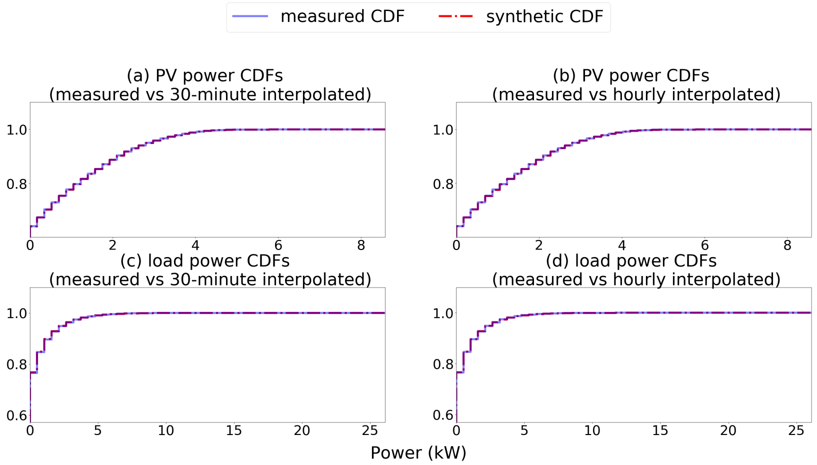

Figure 6.6: CDFs of measured 5-minute PV power data compared to synthetic 5-minute data interpolated from (a) 30-minute power data and (b) hourly power data; CDFs of measured 5-minute load power data compared to synthetic 5-minute data interpolated from (c) 30-minute power data and (d) hourly power data.

Data distribution and autocorrelation

To illustrate the distances between the data probability distributions of measured and synthetic power profiles interpolated by SRAGN, cumulative distribution functions (CDFs) are generated in Figure 6.6 for measured and interpolated 5-minute power datasets. For all the evaluated scenarios, there is almost no visible difference between the synthetic and measured CDFs which indicates that the SRGAN model can generate 5-minute interpolated power profiles from the same data probability distributions of the measured data.

Figure 6.7 demonstrates the mean daily autocorrelation profiles of measured and SRGAN interpolated datasets for all the four evaluated scenarios: PV/load data interpolation from 30-minute/hourly resolution. To compute a mean daily autocorrelation profile, autocorrelations are calculated for all the daily power profiles in the measured/synthetic evaluation set. Then they are averaged for each 5-minute timestamp of a day. Like the CDF results, the mean daily autocorrelations of the SRGAN interpolated data match quite well with the ground truth, which means the SRGAN model can capture the temporal characteristics of 5-minute load and PV power profiles.

Performances in various types of power profiles

It is vital to ensure that the SRGAN model performs well against different types of PV/load power profiles and to assesses what types of PV/load power scenarios

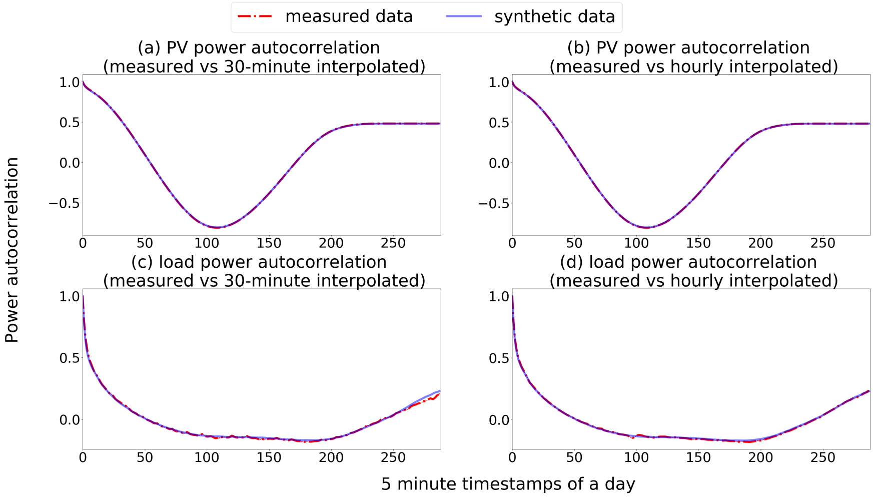

Figure 6.7: Mean daily autocorrelation profiles of measured 5-minute PV power data compared to synthetic 5-minute data interpolated from (a) 30-minute power data and (b) hourly power data; Mean daily autocorrelation profiles of measured 5-minute load power data compared to synthetic 5-minute data interpolated from (c) 30-minute power data and (d) hourly power data

result in better performances. Hence, the PV and load power profiles in the test set are segmented into different clusters and then assessments are carried on these clusters. Daily clearness index is used to separate PV power profiles as it provides a reasonable indication of how clear/cloudy a day is. As the daily clearness index ranges between 0 to 0.8 in the test set, eight equally spaced clearness index intervals of 0.1 are used to group the PV power profiles. K-means algorithm is used to cluster the normalised 30-minute and hourly load power profiles, in this analysis five clusters are adopted for both temporal resolutions. The evaluation metric is the normalised mean squared error (NRMSE) in the daily totals of 5-minute power. The reasons for selecting this metric instead of the JSD used in the model tuning process are two-fold: 1. The interpolated data probability distributions match quite well with the ground truth. As a result the JSDs of various clusters of power profiles are all quite insignificant; 2. This metric is also adopted in a couple of other similar studies [208, 210].

Figure 6.8 demonstrates the NRMSEs in daily PV totals for different ranges of daily clearness index. As the clearness index increases, the NRMSEs decreases for both the 30-minute and hourly interpolated datasets. This is expected as there are more weather transients during cloudy days, making it difficult for the SRGAN model to capture all the uncertainties within the PV power profiles accurately.

Figure 6.9 shows the K-means cluster centroids of 30-minute and hourly load power data and their corresponding NRMSEs in daily totals. Both data

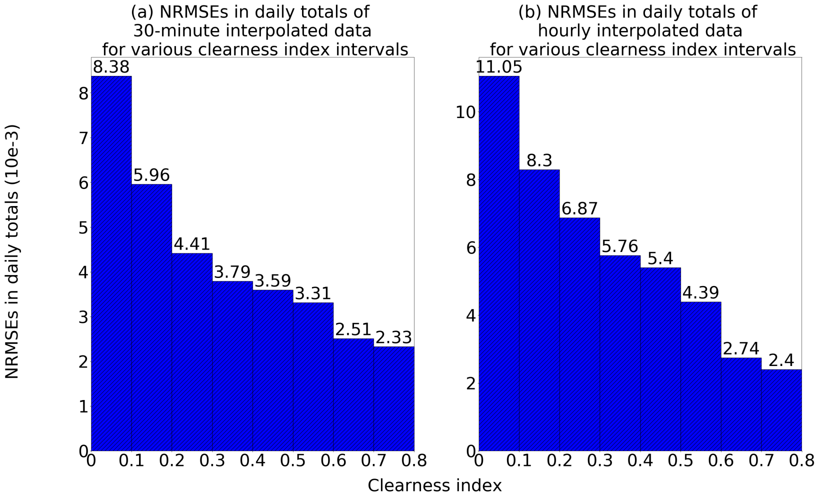

Figure 6.8: NRMSEs in daily totals of 5-minute synthetic data interpolated from (a) 30-minute and (b) hourly PV power data for different clearness index intervals.

granularities end up with similar load clusters. Load profiles with relatively small daytime focused consumption (Cluster 2) result in the smallest NRMSEs in daily totals for both evaluated scenarios, followed by Cluster 5, which contains power profiles with morning and evening peaks. The other three clusters have higher NRMSEs in terms of daily totals, with small differences between them. Overall the NRMSEs are relatively small and stable across various clusters of load profiles, which means the SRGAN model performs well regardless of the type of a load curve.

Performance in a benchmark dataset

It is worthwhile to investigate the performance of the trained SRGAN model on a different dataset. As a widely adopted benchmark dataset, the Smart Grid Smart City (SGSC) dataset includes 30-minute smart meter data (primarily load data) collected between 2010 and 2014 from Australian households in the state of New South Wales (NSW). In this case study, one year data of 2013-2014 is used for validation, including 2839 customers with a full year of load data and 43 households with a whole year of PV data.

Figure 6.10 demonstrates a few daily power profiles of SGSC data and their interpolated 5-minute power profiles using the SRGAN model trained using the Solar Analytics dataset. Although there is no ground truth of 5-minute SGSC data, visually the interpolated 5-minute power profiles seem realistic and contain weather transients and load spikes that can not be observed from the original measured 30-minute profiles. As there is no measured 5-minute data in the

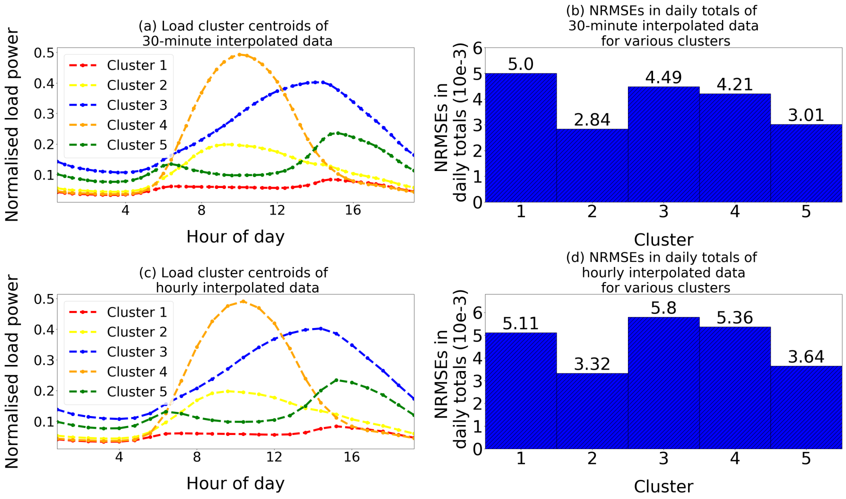

Figure 6.9: Cluster centroids of (a) 30-minute and (c) hourly normalised load power data and NRMSEs in daily totals of 5-minute synthetic data interpolated from (b) 30-minute and (d) hourly PV power data within these clusters.

SGSC dataset, it is not possible to compare the data probability distributions of the measured and interpolated datasets. Instead the adopted metric is the NRMSEs in daily totals, which is already used above for validating the model on different types of power profiles. As a result, the NRMSEs in daily totals of PV power and load power are respectively 0.0039 and 0.0014, which are comparable with the NRMSEs for the test set of SolA dataset (0.0025 for PV and 0.0024 for load). Moreover, the NRMSEs in load daily totals are even lower for the SGSC dataset. Since the SGSC dataset is collected in a different year and quite likely from a different group of households (the Solar Analytics training set only has 693 NSW PV customers and the remaining 1647 PV households are from other states), this shows that the trained SRGAN model is likely to have the same level of performance in other datasets with different time windows and geographical scopes.

6.4.3 Conditional SRGAN

The only input to the SRGAN model is the low-resolution profile without any extra information. However, suppose more information of the power profile can be leveraged to direct the interpolation process of the SRGAN model. In that case, it extends to a conditional SRGAN (CSRAGN) and the interpolation results may be improved. This information Y is also referred to as a class label related to the seasonality or classification of the input power profiles, such as the season/month of a year, load cluster labels. Y can be added to both the generator and the discriminator as an extra input vector. As a result, they are

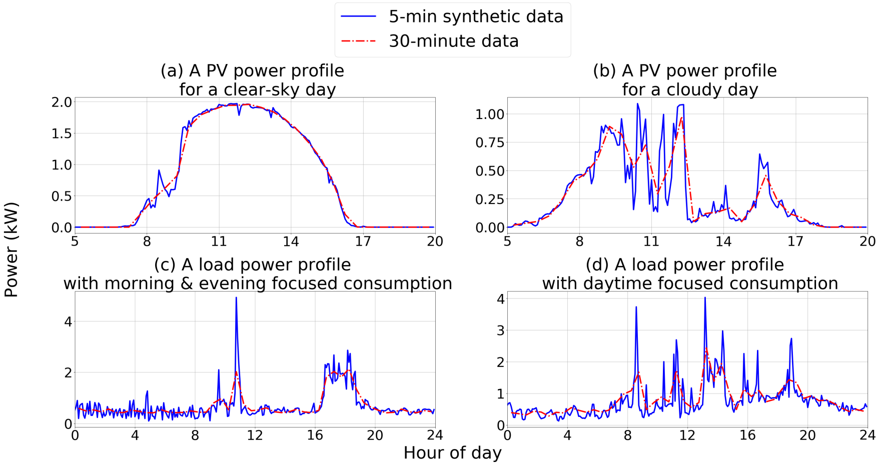

Figure 6.10: 30-minute measured and 5-minute interpolated PV power profiles from the SGSC dataset for (a) a clear-sky day and (b) a cloudy day; 30-minute measured and 5-minute interpolated load power profiles from the SGSC dataset with (c) morning and evening focused consumption and (d) daytime focused consumption.

both conditioned on Y and Eqn. 6.6 can be easily adjusted to the loss function of the CSRGAN model:

$$\min_{G}\max_{D} V(D,G) = \mathbb{E}_{X^{HR}\sim p_{HR}}[\log D(X^{HR}|Y)] + \mathbb{E}_{X^{LR}\sim p_{LR}}[\log(1 - D(G(X^{LR}|Y)))] + \lambda \times (\frac{1}{uM}\sum_{t=1}^{uM}(X_t^{HR} - G(X^{LR})_t)^2) \tag{6.7}$$

In this case study, the SRGAN model is converted to a CSRGAN model by adding month number and cluster label as the extra information for interpolating PV and load power profiles. Month number could help the PV power profile interpolation as it may be related to the seasonal effects on cloud movements and clustering label could also be useful for generating interpolated load profiles as various load clusters may have their distinct load characteristics such as the amount of the consumption spikes. K-means algorithm is applied to cluster the low-resolution load power datasets (30-minute and hourly), Davies-Bouldin index (DBI) is used as the metric to select the optimal numbers of clusters. As a result, 12 clusters are adopted for clustering both the 30-minute and hourly load power data.

An alternative naive prediction model is also implemented as a comparison to the SRGAN and CSRGAN approaches. The main idea of the naive prediction is that for a given 30 minute/hourly daily profile in the evaluation set, another

Table 6.1: NRMSEs in daily/monthly/yearly totals of interpolated 5-minute PV and load power data.

NRMSE in PV totals

Method	input data resolution	Daily	Monthly	Yearly
CSRGAN	30-minute	**0.0025**	**0.0012**	**0.0007**
	hourly	0.0036	**0.0015**	**0.0011**
SRGAN	30-minute	0.0025	0.0017	0.0016
	hourly	**0.0032**	0.0016	0.0012
Naive prediction	30-minute	0.0129	0.0072	0.0063
	hourly	0.0111	0.0057	0.0048

NRMSE in load totals

Method	input data resolution	Daily	Monthly	Yearly
CSRGAN	30-minute	**0.0023**	**0.0015**	**0.0018**
	hourly	0.0039	0.0040	0.0053
SRGAN	30-minute	0.0024	0.0019	0.0023
	hourly	**0.0028**	**0.0017**	**0.0017**
Naive prediction	30-minute	0.0185	0.0127	0.0167
	hourly	0.0172	0.0111	0.0143

daily profile in the training set that has the closest Euclidean distance is selected. Then for predicting the 5-minute profile, the corresponding 5-minute profile of the closest 30-minute/hour profile is adopted as a naive prediction. To make a comprehensive comparison, results are also derived for the cases where measured 30-minute, hourly and 5-minute datasets are available. The results of 5-minute dataset are used as an ideal case which allows us to compute the errors in estimating electricity costs and battery savings, whereas 30-minute and hourly datasets are applied to produce a baseline of the cost and saving results.

Estimation of daily, monthly and yearly totals

Table 6.1 compare the NRMSEs in the interpolated 5-minute daily, monthly and yearly totals using the CSRGAN, SRGAN and naive prediction approaches. For the evaluated scenarios, the CSRGAN approach has a better overall performance in estimating the PV power totals compared to the SRGAN model. The only exception is when predicting the daily PV totals using hourly data as input temporal resolution. On the other hand, inputting additional information only improves the estimation of load power totals when interpolating 30-minute load data, the CSRGAN model results in larger NRMSEs when hourly data is provided.

It is also vital to inspect how the NRMSEs of these interpolation models fluctuate for different households in the test set. Moreover, it would be desirable to compare the NRMSEs against other relevant studies. Although there is no existing studies on interpolating 5-minute PV/load power data, studies in [208–210] interpolate 5-minute/10-minute irradiance data from hourly data

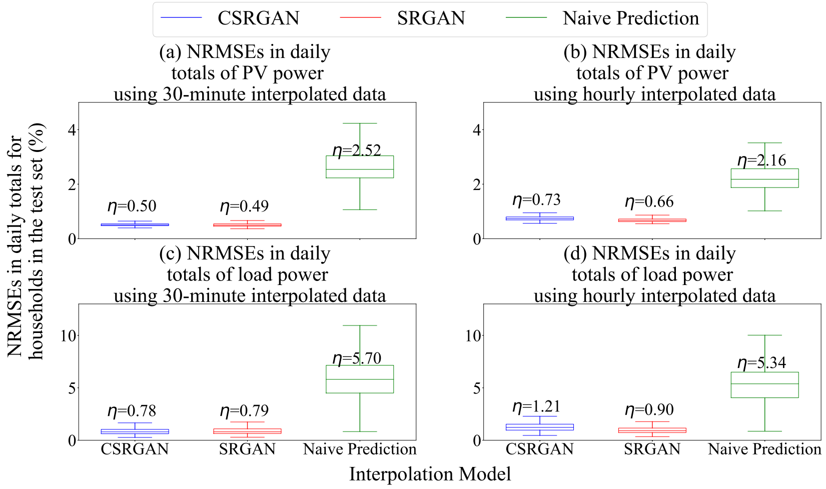

Figure 6.11: Household-level NRMSEs in daily totals of 5-minute PV power data interpolated from (a) 30-minute and (b) hourly measured data, using the CSRGAN, SRGAN and naive prediction approaches; household-level NRMSEs in daily totals of 5-minute load power data interpolated by the CSRGAN, SRGAN and naive prediction approaches, from (c) 30-minute and (d) hourly measured data across households in the test set. η is the median value of the household-level NRMSEs for an interpolation scenario using one interpolation model.

as reviewed in Chapter 2. As PV generation is strongly dependent on solar irradiance data, the performances of our model and the reviewed studies can be roughly compared. It should be noted that the comparisons are not entirely fair as the reviewed studies interpolate irradiance data for a few weather stations while this study aims to interpolate PV power data for households. They reported one NRMSE in the daily totals of solar irradiance for each weather station, taking account of all the collected daily irradiance profiles for that weather station. We use the same metric to generate the box plots in Figure 6.11 for each evaluated scenario and interpolation model. It should be noted that the NRMSEs on this plot are different to the NRMSEs of daily totals in Table 6.1: The NRMSEs in Table 6.1 are computed using all daily profiles in the test set while the household-level NRMSEs in Figure 6.11 are generated individually for each household in the test set to form a box plot, using on year of daily load/PV power profiles.

As the reviewed studies all used small datasets and it is unclear whether these NRMSEs are normally distributed, it makes more sense to compare the medians of the NRMSEs instead of their means. Hence, the medians (η) for each evaluated interpolation scenario are displayed on top of the box plots in Figure 6.11 for each interpolation model. The median NRMSEs in daily totals of irradiance among various reported locations are respectively 3% in [208], 0.65%

in [209] and 0.9% in [210]. Our model has a better performance compared to the approach in [210]. The work in [208] and [209] interpolate 10-minute instead of 5-minute data from hourly measurements. Despite having a higher upsampling factor, the NRMSE median shown in Figure 6.11(b) is 0.66 % for the SRGAN model, which is quite close to the 0.65 % median in [209] and much smaller compared to the reported value in [208].

Similar to the results in Table 6.1, the SRGAN model has better performances over the other two alternative methods when interpolating 5-minute load/PV data from hourly resolution. However, the medians and interquartile ranges (IQR) of the NRMSEs across test set households for interpolating PV and load power data are relatively close between the SRGAN and CSRGAN models when using 30-minute data as inputs. Hence, paired Wilcoxon signed-rank tests are performed, which is a non-parametric statistical significance test to compare two paired groups of samples [219]. In this case, we use one-sided instead of two-sided tests to assess which approach results in smaller NRMSEs for households in the test set. Another aim is to determine whether there are sufficient households in the test set for us to find the optimal interpolation approach for each evaluated scenario. As a result, Wilcoxon signed-rank tests are conducted to compare the three approaches for each considered interpolation scenario, all of them returned a p-value < 0.05. Moreover, the statistical tests show that the CSGAN model achieves the lowest NRMSEs (p-value $= 0.0012$) in terms of interpolating load data from 30-minute resolution. On the other hand, despite having a lower NRMSE in Table 6.1, the CSRGAN model leads to higher household-level NRMSEs (p-value $= 0.006$) for interpolating 5-minute PV power data from 30-minute resolution compared to the SRGAN model.

Unfortunately, the metric reported in the reviewed load data interpolation study [211] was the root mean square error (RMSE), and there was no unit provided for the RMSEs. Moreover, the reviewed study aimed to interpolate very high frequency data (100/1000 Hz), which is quite different to our scope. Hence, it is not feasible to compare between our model and the approach in [211].

Estimation of electricity costs and battery saving potentials

One potential end-use application of the interpolated PV/load data is to provide more accurate estimations of electricity costs and battery saving potentials for households with PV when only coarse meter data is available. In this section, the battery simulation model in Chapter 4 is adopted to evaluate the interpolated data which requires PV and load data as inputs, simulates the operations of a residential battery and computes the electricity costs with & without a battery and potential battery savings for an Australian solar household. Also, this case study follows the same economic parameters, battery specifications, charging & discharging algorithm and tariff structures (flat and time-of-use (ToU)) in Chapter 4. For each household, the battery size range is set to be 1-15 kWh with an increment of 1 kWh, the potential battery savings are computed for each battery size by taking the difference between the electricity costs with &

Table 6.2: NRMSEs and r-squared values for estimating yearly electricity costs and battery saving potentials using low-resolution measured data, 5-minute data interpolated by the CSRGAN, SRGAN and naive prediction models.

Errors in yearly electricity costs

Tariff		Flat		ToU	
Method	input data resolution	NRMSE	r squared	NRMSE	r squared
CSRGAN	**30-minute**	**0.00244**	**0.99972**	**0.00242**	**0.99976**
	hourly	0.00457	0.99903	0.00478	0.99906
SRGAN	30-minute	0.00275	0.99965	0.00313	0.99960
	hourly	**0.00284**	**0.99963**	**0.00287**	**0.99966**
Measured	30-minute	0.00293	0.99960	0.00309	0.99961
	hourly	0.00483	0.99892	0.00519	0.99890
Naive prediction	30-minute	0.02161	0.97842	0.02336	0.97766
	hourly	0.01843	0.98430	0.01940	0.98459

Errors in yearly battery savings

Tariff		Flat		ToU	
Method	input data resolution	NRMSE	r squared	NRMSE	r squared
CSRGAN	**30-minute**	**0.02589**	**0.96576**	**0.01822**	**0.98621**
	hourly	0.03861	0.92970	0.02844	0.96887
SRGAN	30-minute	0.02927	0.95917	0.02097	0.98220
	hourly	**0.03664**	**0.93686**	**0.02520**	**0.97526**
Measured	30-minute	0.04126	0.93673	0.02673	0.97724
	hourly	0.06412	0.84935	0.04320	0.94229
Naive prediction	30-minute	0.05826	0.87075	0.04926	0.92733
	hourly	0.05752	0.87180	0.04612	0.93373

without a battery.

Table 6.2 illustrates the normalised root mean squared error (NRMSE) and r-squared values in estimated yearly electricity costs and battery saving potentials using low-resolution measured data and interpolated 5-minute data for the households in the test set. For using 30-minute PV & load data as inputs, the CSRGAN model is able to achieve the smallest errors in estimating electricity costs and battery saving potentials under the tested flat and ToU tariffs. On the other hand, in terms of adopting hourly PV & load data, the SRGAN approach produces the smallest NRMSEs and the highest r-squared values in estimating electricity costs and battery savings for all the evaluated scenarios. Both the CSRGAN and SRGAN have much better performances than the measured low-resolution data and the naive prediction approach. Compared to the baseline approach of using hourly measured data, under the flat and the ToU tariffs, the SRGAN model respectively leads to 41.2% and 44.7% error reductions in estimating electricity costs, 42.9% and 41.7% error reductions in estimating battery saving potentials. This indicates that these two models can potentially address the inaccuracies in the estimated costs and savings caused by using low granularity data in the power optimisation of PV battery systems.

6.5 Summary

In this chapter, a SRGAN based model is proposed to synthetically interpolate 5-minute average PV and load power data from 30-minute and hourly data. Evaluations on the developed model show that the SRGAN model can fully capture the data probability distribution and temporal characteristics of the measured 5-minute data, as shown in Figure 6.6 and Figure 6.7. The results in Figure 6.10 and NRMSEs in daily totals also indicate that even though the SRGAN model is trained using the Solar Analytics dataset, it achieves the same level of performances on the SGSC dataset, which has a different time window and geographical scope. Moreover, the results in Table 6.2 illustrate that the SRGAN interpolated data can be applied to derive much better estimations of electricity costs and battery saving potentials of PV battery systems, than using low-resolution data or a naive forecasting approach. This indicates that the proposed model can address the issue of limited proprietary high-resolution data in modelling and optimisation of a PV-integrated battery system.

Chapter 7

Conclusion

The lack of high-resolution proprietary load and PV data poses challenges on modelling and optimisation of PV-battery systems.

This thesis addressed this problem by firstly conducting a sensitivity analysis to investigate the impacts of various data granularities and battery efficiency settings on the optimised costs of PV battery scheduling models. We concluded 5-minute temporal resolution is sufficient to compute results with a good level of accuracy and set it as the targeted temporal resolution for the remaining part of the thesis.

Moreover, this thesis developed a data extrapolation method using net meter energy data clustering, which can convert a limited amount of input 5-minute net/gross meter energy data to a whole year of energy data that still produces accurate results in a battery size optimisation model.

Furthermore, a DCGAN based model was presented in this thesis to generate high-quality 5-minute synthetic residential PV and load power data from random noises. The proposed model was also used in a battery size optimisation model, which can estimate electricity costs and perform energy storage sizing for new residential customers with no historical data.

Lastly, a SRGAN based approach was proposed in this thesis which can synthetically interpolate 5-minute PV and load power data from lower granularity (30-minute/hourly) PV and load power data. Statistically, the interpolated data almost makes no difference to the measured data and it significantly reduces the inaccuracies caused by using coarse data in a PV-battery optimisation model.

For future work, as only a RB and a LP model are included with a single objective of minimising electricity costs in our temporal resolution sensitivity analysis, it would be worthwhile to evaluate the impacts of temporal resolutions in other optimisation models such as dynamic programming, quadratic programming, mixed integer linear programming, evolutionary algorithms and reinforcement learning. Other optimisation objectives such as reducing peak demands, minimising battery degradation can also be considered for a more detailed granularity sensitivity study. In terms of battery efficiency and SOC tracking, there is a need to extract more input features for our existing model or to develop more advanced non-linear models to improve efficiency and SOC estimations in a PV battery optimisation model.

As only a single optimisation objective of maximising self-consumption is used in Chapter 4, it would be interesting to see how well the data extrapolation model based on net meter data clustering based could perform for other optimisation goals such as battery degradation reduction, peak demand reduction or price arbitrage. The dataset used is from solar customers in Australia so it would be worthwhile to apply the approach to a dataset with customers in other countries to see how well this approach generalises in a different region. The temporal resolution adopted in Chapter 4 is half-hour net meter energy data. It could be interesting to explore what data granularity optimises the trade-offs between computations and performances of our proposed model.

It would also be worthwhile to extend and validate the DCGAN model proposed in Chapter 5 on different geographical scopes or commercial sites. Moreover, in Chapter 5, the considered temporal resolution is 5-minute data and the sample profile horizon is 1 day. It will be interesting to see how feasible the proposed DCGAN approach can be adjusted to generate power profiles with different granularities and sample lengths or for a different renewable resource such as wind generation. The adopted conditional DCGAN model only uses a single sample label for training, although it is feasible to include multiple labels. As in practice, more demographic and lifestyle information of a household could be potentially obtained, it will be desirable to explore how multiple labels can be incorporated during the data synthesis process.

Chapter 6 has explored providing additional information during the data interpolation process, which turns the SRGAN model into a CRGAN approach. The CSRGAN model results in superior performance for interpolating 30-minute power data. However, it leads to more inaccuracies when interpolating hourly data, especially for load data. Hence, it would be desirable for future work to improve the CSRGAN model and explore other types of information that could assist the interpolation process. It will also be worthwhile to evaluate the proposed approach for interpolating power profiles of other types of renewable generation (e.g. wind) or finer temporal resolutions (e.g. 1-minute), in order to assess how well the SRGAN model generalises in time series that are different to the datasets adopted in this work.

Bibliography

[1] Bloomberg NEF. *Energy, Vehicles, Sustainability 10 Predictions for 2020.* https://about.bnef.com/blog/energy-vehicles-sustainability-10-predictions-for-2020/. [Online; accessed 10-April-2020]. 2020.

[2] Solar Power Europe. "Global Market Outlook For Solar Power/2020–2024". In: *Solar Power Europe: Brussels, Belgium* (2020).

[3] Lavinia Poruschi, Christopher L. Ambrey, and James C.R. Smart. "Revisiting feed-in tariffs in Australia: A review". In: *Renewable and Sustainable Energy Reviews* 82.October 2016 (2018), pp. 260–270. ISSN: 18790690. DOI: 10.1016/j.rser.2017.09.027.

[4] Renewable Energy Policy for the 21st Century (REN21). *Renewables 2017 Global Status Report.* http://www.ren21.net/gsr-2017/. [Online; accessed 15-June-2018]. 2017.

[5] F. J. Ramarez et al. "Combining feed-in tariffs and net-metering schemes to balance development in adoption of photovoltaic energy: Comparative economic assessment and policy implications for European countries". English. In: *Energy Policy* 102 (2017), pp. 440–452.

[6] Smart Energy Council (SEC). *Australian Energy Storage Market Analysis.* https://www.smartenergy.org.au/sites/default/files/uploaded-content/field_f_content_file/australian_energy_storage_market_analysis_report_sep18_final.pdf. [Online; accessed 15-September-2018]. 2018.

[7] Global Sustainable Energy Solution. *Grid-Connected PV Systems with Battery Storage.* 2015.

[8] NSW Department of Industry, Resources & Energy. *Metering Fact Sheet.* http://www.resourcesandenergy.nsw.gov.au/__data/assets/pdf_file/0010/683506/sbs-meteringfactsheet.PDF. [Online; accessed 21-November-2016]. 2016.

[9] B. Yildiz et al. "Recent advances in the analysis of residential electricity consumption and applications of smart meter data". In: *Applied Energy* (2017). ISSN: 03062619. DOI: 10.1016/j.apenergy.2017.10.014. URL: http://linkinghub.elsevier.com/retrieve/pii/S0306261917314265.

[10] Sangeetha Chandrashekeran. *Smart electricity meters are here, but more is needed to make them useful to customers.* http://theconversation.com/smart-electricity-meters-are-here-but-more-is-needed-to-make-them-useful-to-customers-92029. [Online; accessed 26-August-2019]. 2018.

[11] T. Beck et al. “Assessing the influence of the temporal resolution of electrical load and PV generation profiles on self-consumption and sizing of PV-battery systems”. In: *Applied Energy* 173 (2016), pp. 331–342. ISSN: 03062619. DOI: 10.1016/j.apenergy.2016.04.050. URL: http://dx.doi.org/10.1016/j.apenergy.2016.04.050.

[12] Yi Wang et al. “Review of Smart Meter Data Analytics: Applications, Methodologies, and Challenges”. In: *IEEE Transactions on Smart Grid* 10.3 (2019), pp. 3125–3148. ISSN: 19493053. DOI: 10.1109/TSG.2018.2818167. arXiv: 1802.04117.

[13] A. Cervone et al. “Optimization of the battery size for PV systems under regulatory rules using a Markov-Chains approach”. In: *Renewable Energy* 85 (2016), pp. 657–665. ISSN: 18790682. DOI: 10.1016/j.renene.2015.07.007. URL: http://dx.doi.org/10.1016/j.renene.2015.07.007.

[14] Ian Richardson and Murray Thomson. “Integrated simulation of photovoltaic micro-generation and domestic electricity demand: A one-minute resolution open-source model”. In: *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy* 227.1 (2013), pp. 73–81. ISSN: 09576509. DOI: 10.1177/0957650912454989.

[15] Paul A. Gagniuc. *Markov chains : from theory to implementation and experimentation.* John Wiley and Sons, 2017.

[16] F. McLoughlin, A. Duffy, and M. Conlon. “The Generation of Domestic Electricity Load Profiles through Markov Chain Modelling”. In: *3rd International Scientific Conference on Energy and Climate Change* (2010), pp. 18–27. URL: https://arrow.dit.ie/dubencon2.

[17] Hongjie Jia, Yunfei Mu, and Yan Qi. “A statistical model to determine the capacity of battery-supercapacitor hybrid energy storage system in autonomous microgrid”. In: *International Journal of Electrical Power and Energy Systems* 54 (2014), pp. 516–524. ISSN: 01420615. DOI: 10.1016/j.ijepes.2013.07.025. URL: http://dx.doi.org/10.1016/j.ijepes.2013.07.025.

[18] Ian Richardson et al. “Domestic electricity use: A high-resolution energy demand model”. In: *Energy and Buildings* 42.10 (2010), pp. 1878–1887. ISSN: 03787788. DOI: 10.1016/j.enbuild.2010.05.023. URL: http://dx.doi.org/10.1016/j.enbuild.2010.05.023.

[19] Jacopo Torriti. “A review of time use models of residential electricity demand”. In: *Renewable and Sustainable Energy Reviews* 37 (2014), pp. 265–272. ISSN: 13640321. DOI: 10.1016/j.rser.2014.05.034. URL: http://dx.doi.org/10.1016/j.rser.2014.05.034.

[20] Chun Liu et al. "Generative adversarial networks and convolutional neural networks based weather classification model for day ahead short-term photovoltaic power forecasting". In: *Energy Conversion and Management* 181.August 2018 (2018), pp. 443–462. ISSN: 01968904. DOI: 10.1016/j.enconman.2018.11.074.

[21] Yize Chen et al. "Model-Free Renewable Scenario Generation Using Generative Adversarial Networks". In: *IEEE Transactions on Power Systems* 33.3 (2017), pp. 3265–3275. ISSN: 0885-8950. DOI: 10.1109/TPWRS.2018.2794541. arXiv: 1707.09676. URL: http://arxiv.org/abs/1707.09676.

[22] Rui Tang et al. "Generating Residential PV Production and Electricity Consumption Scenarios via Generative Adversarial Networks". In: *2018 Asia-Pacific Solar Research Conference* (2018). URL: http://apvi.org.au/solar-research-conference/wp-content/uploads/2019/01/068_DI_Tang_R_2018.pdf.

[23] Thomas Ackermann, Göran Andersson, and Lennart Söder. "Distributed generation: a definition". In: *Electric power systems research* 57.3 (2001), pp. 195–204.

[24] Ying Yi Hong and Jie Kai Lin. "Interactive multi-objective active power scheduling considering uncertain renewable energies using adaptive chaos clonal evolutionary programming". In: *Energy* 53 (2013), pp. 212–220. ISSN: 03605442. DOI: 10.1016/j.energy.2013.02.070. URL: http://dx.doi.org/10.1016/j.energy.2013.02.070.

[25] Bo Lu and Mohammad Shahidehpour. "Short-term scheduling of battery in a grid-connected PV/battery system". In: *IEEE Transactions on Power Systems* 20.2 (2005), pp. 1053–1061. ISSN: 08858950. DOI: 10.1109/TPWRS.2005.846060.

[26] Akihiro Yoza et al. "Optimal capacity and expansion planning methodology of PV and battery in smart house". In: *Renewable Energy* 69 (2014), pp. 25–33. ISSN: 09601481. DOI: 10.1016/j.renene.2014.03.030. URL: http://dx.doi.org/10.1016/j.renene.2014.03.030.

[27] Yang Zhang et al. "Comparative study of hydrogen storage and battery storage in grid connected photovoltaic system: Storage sizing and rule-based operation". In: *Applied Energy* 201 (2017), pp. 397–411. ISSN: 03062619. DOI: 10.1016/j.apenergy.2017.03.123. URL: http://dx.doi.org/10.1016/j.apenergy.2017.03.123.

[28] Roberto Romano et al. "Combined Operation of Electrical Loads, Air Conditioning and Photovoltaic-Battery Systems in Smart Houses". In: *Applied Sciences* 7.6 (2017), p. 525. ISSN: 2076-3417. DOI: 10.3390/app7050525. URL: http://www.mdpi.com/2076-3417/7/5/525.

[29] Ce Shang, Dipti Srinivasan, and Thomas Reindl. “Generation-scheduling-coupled battery sizing of stand-alone hybrid power systems”. In: *Energy* 114 (2016), pp. 671–682. ISSN: 03605442. DOI: 10.1016/j.energy.2016.07.123. URL: http://dx.doi.org/10.1016/j.energy.2016.07.123.

[30] Tarek AlSkaif et al. “Reputation-based joint scheduling of households appliances and storage in a microgrid with a shared battery”. In: *Energy and Buildings* 138 (2017), pp. 228–239. ISSN: 03787788. DOI: 10.1016/j.enbuild.2016.12.050. URL: http://dx.doi.org/10.1016/j.enbuild.2016.12.050.

[31] M. Castillo-Cagigal et al. “A semi-distributed electric demand-side management system with PV generation for self-consumption enhancement”. In: *Energy Conversion and Management* 52.7 (2011), pp. 2659–2666. ISSN: 01968904. DOI: 10.1016/j.enconman.2011.01.017. URL: http://dx.doi.org/10.1016/j.enconman.2011.01.017.

[32] Yumiko Iwafune et al. “Cooperative home energy management using batteries for a photovoltaic system considering the diversity of households”. In: *Energy Conversion and Management* 96 (2015), pp. 322–329. ISSN: 01968904. DOI: 10.1016/j.enconman.2015.02.083. URL: http://dx.doi.org/10.1016/j.enconman.2015.02.083.

[33] Ren-Shiou Liu. “An Algorithmic Game Approach for Demand Side Management in Smart Grid with Distributed Renewable Power Generation and Storage”. In: *Energies* 9.8 (2016), p. 654. ISSN: 1996-1073. DOI: 10.3390/en9080654. URL: http://www.mdpi.com/1996-1073/9/8/654.

[34] Guido Lorenzi and Carlos Augusto Santos Silva. “Comparing demand response and battery storage to optimize self-consumption in PV systems”. In: *Applied Energy* 180 (2016), pp. 524–535. ISSN: 03062619. DOI: 10.1016/j.apenergy.2016.07.103. URL: http://dx.doi.org/10.1016/j.apenergy.2016.07.103.

[35] Elizabeth L. Ratnam, Steven R. Weller, and Christopher M. Kellett. “Central versus localized optimization-based approaches to power management in distribution networks with residential battery storage”. In: *International Journal of Electrical Power and Energy Systems* 80 (2016), pp. 396–406. ISSN: 01420615. DOI: 10.1016/j.ijepes.2016.01.048. URL: http://dx.doi.org/10.1016/j.ijepes.2016.01.048.

[36] Fei Yang and Xiaohua Xia. “Techno-economic and environmental optimization of a household photovoltaic-battery hybrid power system within demand side management”. In: *Renewable Energy* 108 (2017), pp. 132–143. ISSN: 09601481. DOI: 10.1016/j.renene.2017.02.054. URL: http://linkinghub.elsevier.com/retrieve/pii/S0960148117301404.

[37] Akihiro Yoza et al. “Optimal scheduling method of controllable loads in smart house considering forecast error”. In: *Proceedings of the International Conference on Power Electronics and Drive Systems* (2013), pp. 84–89. DOI: 10.1109/PEDS.2013.6526993.

[38] Yasuaki Miyazato et al. “Multi-Objective Optimization for Smart House Applied Real Time Pricing Systems”. In: *Sustainability* 8.12 (2016), p. 1273. ISSN: 2071-1050. DOI: 10.3390/su8121273. URL: http://www.mdpi.com/2071-1050/8/12/1273.

[39] Zhengen Ren, George Grozev, and Andrew Higgins. “Modelling impact of PV battery systems on energy consumption and bill savings of Australian houses under alternative tariff structures”. In: *Renewable Energy* 89 (2016), pp. 317–330. ISSN: 18790682. DOI: 10.1016/j.renene.2015.12.021. URL: http://dx.doi.org/10.1016/j.renene.2015.12.021.

[40] E. Georges, J. E. Braun, and V. Lemort. “A general methodology for optimal load management with distributed renewable energy generation and storage in residential housing”. In: *Journal of Building Performance Simulation* 10.2 (2017), pp. 224–241. ISSN: 1940-1493. DOI: 10.1080/19401493.2016.1211738. URL: https://www.tandfonline.com/doi/full/10.1080/19401493.2016.1211738.

[41] S. Surender Reddy. “Optimal power flow with renewable energy resources including storage”. In: *Electrical Engineering* 99.2 (2016), pp. 1–11. ISSN: 14320487. DOI: 10.1007/s00202-016-0402-5.

[42] K. Kusakana. “Daily operation cost minimization of photovoltaic-diesel-battery hybrid systems using different control strategies”. In: *IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society* (2016), pp. 3609–3613. DOI: 10.1109/IECON.2015.7392661.

[43] Rn Mahanty and Pbd Gupta. “Short-term real-power scheduling considering fuzzy factors in an autonomous system using genetic algorithms”. In: *IEE Proceedings-Generation, Transmission and . . .* 151.3 (2004), pp. 201–212. ISSN: 1350-2360. DOI: 10.1049/ip-gtd. URL: http://digital-library.theiet.org/content/journals/10.1049/ip-gtd{_}20040098.

[44] Eric J. Hoevenaars and Curran A. Crawford. “Implications of temporal resolution for modeling renewables-based power??systems”. In: *Renewable Energy* 41 (2012), pp. 285–293. ISSN: 09601481. DOI: 10.1016/j.renene.2011.11.013. URL: http://dx.doi.org/10.1016/j.renene.2011.11.013.

[45] D. P. Jenkins, J. Fletcher, and D. Kane. “Model for evaluating impact of battery storage on microgeneration systems in dwellings”. In: *Energy Conversion and Management* 49.8 (2008), pp. 2413–2424. ISSN: 01968904. DOI: 10.1016/j.enconman.2008.01.011.

[46] Henning Tischer and Gregor Verbic. “Towards a smart home energy management system - A dynamic programming approach”. In: *2011 IEEE PES Innovative Smart Grid Technologies* (2011), pp. 1–7. DOI: 10.1109/ISGT-Asia.2011.6167090. URL: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp={\&}arnumber=6167090{\&}contentType=Conference+Publications{\&}searchField{\%}3DSearch{_}All{\%}26queryText{\%}3Delectrical+car+battery+performance.

[47] Juan P. Torreglosa et al. “Energy dispatching based on predictive controller of an off-grid wind turbine/photovoltaic/hydrogen/battery hybrid system”. In: *Renewable Energy* 74 (2015), pp. 326–336. ISSN: 09601481. DOI: 10.1016/j.renene.2014.08.010.

[48] Tanguy Hubert and Santiago Grijalva. “Modeling for residential electricity optimization in dynamic pricing environments”. In: *IEEE Transactions on Smart Grid* 3.4 (2012), pp. 2224–2231. ISSN: 19493053. DOI: 10.1109/TSG.2012.2220385.

[49] Khalid Abdulla et al. “Optimal Operation of Energy Storage Systems Considering Forecasts and Battery Degradation”. In: *IEEE Transactions on Smart Grid* PP.99 (2016), pp. 1–11. ISSN: 19493053. DOI: 10.1109/TSG.2016.2606490.

[50] Khalid Abdulla et al. “The Importance of Temporal Resolution in Evaluating Residential Energy Storage”. In: *Ieee Pesgm 2017* (2017), p. 7. URL: https://www.researchgate.net/publication/313857976{_}The{_}Importance{_}of{_}Temporal{_}Resolution{_}in{_}Evaluating{_}Residential{_}Energy{_}Storage.

[51] Christopher J. Bennett, Rodney A. Stewart, and Jun Wei Lu. “Development of a three-phase battery energy storage scheduling and operation system for low voltage distribution networks”. In: *Applied Energy* 146 (2015), pp. 122–134. ISSN: 03062619. DOI: 10.1016/j.apenergy.2015.02.012. URL: http://dx.doi.org/10.1016/j.apenergy.2015.02.012.

[52] Felix Braam et al. “Peak shaving with photovoltaic-battery systems”. In: *IEEE PES Innovative Smart Grid Technologies Conference Europe* 2015-Janua.January (2015), pp. 1–5. DOI: 10.1109/ISGTEurope.2014.7028748.

[53] Danilo Fuselli et al. “Action dependent heuristic dynamic programming for home energy resource scheduling”. In: *International Journal of Electrical Power and Energy Systems* 48.1 (2013), pp. 148–160. ISSN: 01420615. DOI: 10.1016/j.ijepes.2012.11.023. URL: http://dx.doi.org/10.1016/j.ijepes.2012.11.023.

[54] Mohsen Gitizadeh and Hamid Fakharzadegan. “Effects of electricity tariffs on optimal battery energy storage sizing in residential PV/storage systems”. In: *2013 International Conference on Energy Efficient Technologies for Sustainability, ICEETS 2013* (2013), pp. 1072–1077. DOI: 10.1109/ICEETS.2013.6533536.

[55] R. Hanna et al. “Energy dispatch schedule optimization for demand charge reduction using a photovoltaic-battery storage system with solar forecasting”. In: *Solar Energy* 103 (2014), pp. 269–287. ISSN: 0038092X. DOI: 10.1016/j.solener.2014.02.020. URL: http://dx.doi.org/10.1016/j.solener.2014.02.020.

[56] Chanaka Keerthisinghe, Gregor Verbic, and Archie C. Chapman. "Evaluation of a multi-stage stochastic optimisation framework for energy management of residential PV-storage systems". In: *2014 Australasian Universities Power Engineering Conference, AUPEC 2014 - Proceedings* September 2016 (2014). DOI: 10.1109/AUPEC.2014.6966552.

[57] Chanaka Keerthisinghe, Gregor Verbic, and Archie C. Chapman. "A Fast Technique for Smart Home Management: ADP with Temporal Difference Learning". In: *IEEE Transactions on Smart Grid* 3053.c (2016), pp. 1–1. ISSN: 1949-3053. DOI: 10.1109/TSG.2016.2629470. URL: http://ieeexplore.ieee.org/document/7745930/.

[58] Jiahao Li and Michael A. Danzer. "Optimal charge control strategies for stationary photovoltaic battery systems". In: *Journal of Power Sources* 258 (2014), pp. 365–373. ISSN: 03787753. DOI: 10.1016/j.jpowsour.2014.02.066. URL: http://dx.doi.org/10.1016/j.jpowsour.2014.02.066.

[59] Bing Liu et al. "A MPC operation method for a photovoltaic system with batteries". In: *IFAC-PapersOnLine* 28.8 (2015), pp. 807–812. ISSN: 24058963. DOI: 10.1016/j.ifacol.2015.09.068. URL: http://dx.doi.org/10.1016/j.ifacol.2015.09.068.

[60] Mengjun Ming et al. "Multi-Objective Optimization of Hybrid Renewable Energy System Using an Enhanced Multi-Objective Evolutionary Algorithm". In: *Energies* 10.5 (2017), pp. 5–9. ISSN: 1996-1073. DOI: 10.1109/ICEI.2017.27.

[61] A. Nottrott, J. Kleissl, and B. Washom. "Energy dispatch schedule optimization and cost benefit analysis for grid-connected, photovoltaic-battery storage systems". In: *Renewable Energy* 55 (2013), pp. 230–240. ISSN: 09601481. DOI: 10.1016/j.renene.2012.12.036. arXiv: arXiv:1011.1669v3. URL: http://dx.doi.org/10.1016/j.renene.2012.12.036.

[62] Amparo Nunez-Reyes et al. "Optimal scheduling of grid-connected PV plants with energy storage for integration in the electricity market". In: *Solar Energy* 144 (2017), pp. 502–516. ISSN: 0038092X. DOI: 10.1016/j.solener.2016.12.034.

[63] Balint D. Olaszi and Jozsef Ladanyi. "Comparison of different discharge strategies of grid-connected residential PV systems with energy storage in perspective of optimal battery energy storage system sizing". In: *Renewable and Sustainable Energy Reviews* 75.September 2016 (2017), pp. 710–718. ISSN: 18790690. DOI: 10.1016/j.rser.2016.11.046. URL: http://dx.doi.org/10.1016/j.rser.2016.11.046.

[64] H Pezeshki et al. "A Model Predictive Approach for Community Battery Energy Storage System Optimization". In: *PES General Meeting | Conference & Exposition, 2014 IEEE* (2014), pp. 27–31.

[65] Iromi Ranaweera and Ole Morten Midtgård. "Optimization of operational cost for a grid-supporting PV system with battery storage". In: *Renewable Energy* 88 (2016), pp. 262–272. ISSN: 18790682. DOI: 10.1016/j.renene.2015.11.044. URL: http://dx.doi.org/10.1016/j.renene.2015.11.044.

[66] Iromi Ranaweera, Ole-Morten Midtgård, and Magnus Korpås. "Distributed control scheme for residential battery energy storage units coupled with PV systems". In: *Renewable Energy* 113 (2017), pp. 1099–1110. ISSN: 09601481. DOI: 10.1016/j.renene.2017.06.084. URL: http://linkinghub.elsevier.com/retrieve/pii/S0960148117305888.

[67] Elizabeth L. Ratnam, Steven R. Weller, and Christopher M. Kellett. "An optimization-based approach to scheduling residential battery storage with solar PV: Assessing customer benefit". In: *Renewable Energy* 75 (2015), pp. 123–134. ISSN: 18790682. DOI: 10.1016/j.renene.2014.09.008. URL: http://dx.doi.org/10.1016/j.renene.2014.09.008.

[68] Elizabeth L. Ratnam, Steven R. Weller, and Christopher M. Kellett. "Scheduling residential battery storage with solar PV: Assessing the benefits of net metering". In: *Applied Energy* 155 (2015), pp. 881–891. ISSN: 03062619. DOI: 10.1016/j.apenergy.2015.06.061. URL: http://dx.doi.org/10.1016/j.apenergy.2015.06.061.

[69] Y Riffonneau et al. "Optimal Power Flow Management for Grid Connected PV Systems With Batteries". In: *IEEE Transactions on Sustainable Energy* 2.3 (2011), pp. 309–320. ISSN: 1949-3029. DOI: 10.1109/TSTE.2011.2114901.

[70] Luigi Schibuola, Massimiliano Scarpa, and Chiara Tambani. "Influence of charge control strategies on electricity import/export in battery-supported photovoltaic systems". In: *Renewable Energy* 113 (2017), pp. 312–328. ISSN: 09601481. DOI: 10.1016/j.renene.2017.05.089. URL: http://linkinghub.elsevier.com/retrieve/pii/S0960148117304834.

[71] Ryota Suzuki. "Determination Method of Optimal Planning and Operation for Residential PV System and Storage Battery Based on Weather Forecast". In: December (2012), pp. 2–5.

[72] Jen Hao Teng et al. "Optimal charging/discharging scheduling of battery storage systems for distribution systems interconnected with sizeable PV generation systems". In: *IEEE Transactions on Power Systems* 28.2 (2013), pp. 1425–1433. ISSN: 08858950. DOI: 10.1109/TPWRS.2012.2230276.

[73] Trudie Wang, Haresh Kamath, and Steve Willard. "Control and optimization of grid-tied photovoltaic storage systems using model predictive control". In: *IEEE Transactions on Smart Grid* 5.2 (2014), pp. 1010–1017. ISSN: 19493053. DOI: 10.1109/TSG.2013.2292525.

[74] Zhou Wu, Henerica Tazvinga, and Xiaohua Xia. “Optimal Schedule of Photovoltaic-Battery Hybrid System at Demand Side”. In: 2014.December (2014), pp. 10–12.

[75] G. R. Aghajani, H. A. Shayanfar, and H. Shayeghi. “Presenting a multi-objective generation scheduling model for pricing demand response rate in micro-grid energy management”. In: *Energy Conversion and Management* 106 (2015), pp. 308–321. ISSN: 01968904. DOI: `10.1016/j.enconman.2015.08.059`. URL: `http://dx.doi.org/10.1016/j.enconman.2015.08.059`.

[76] A K Barnes, J C Balda, and ... “A semi-Markov model for control of energy storage in utility grids and microgrids with PV generation”. In: *IEEE Transactions on . . .* 6.2 (2015), pp. 546–556. URL: `http://ieeexplore.ieee.org/abstract/document/7045576/`.

[77] Wei-che Chang, Ming-yang Cheng, and Hong-jin Tsai. “Optimization of a Grid-Tied Microgrid Configuration using Dual Storage Systems”. In: Iccas (2013), pp. 1143–1148.

[78] Anderson Hoke et al. “Look-ahead economic dispatch of microgrids with energy storage, using linear programming”. In: *2013 1st IEEE Conference on Technologies for Sustainability (SusTech)* (2013), pp. 154–161. DOI: `10.1109/SusTech.2013.6617313`.

[79] Sam Koohi-Kamali, N. A. Rahim, and H. Mokhlis. “Smart power management algorithm in microgrid consisting of photovoltaic, diesel, and battery storage plants considering variations in sunlight, temperature, and load”. In: *Energy Conversion and Management* 84 (2014), pp. 562–582. ISSN: 01968904. DOI: `10.1016/j.enconman.2014.04.072`. URL: `http://dx.doi.org/10.1016/j.enconman.2014.04.072`.

[80] R Leo, R S Milton, and S Sibi. “Reinforcement learning for optimal energy management of a solar microgrid”. In: *2014 IEEE Global Humanitarian Technology Conference - South Asia Satellite (GHTC-SAS)* (2014), pp. 183 –8. DOI: `10.1109/GHTC-SAS.2014.6967580`. URL: `http://dx.doi.org/10.1109/GHTC-SAS.2014.6967580`.

[81] Adriana Luna et al. “Optimal power scheduling for a grid-connected hybrid PV-wind-battery microgrid system”. In: *Conference Proceedings - IEEE Applied Power Electronics Conference and Exposition - APEC* 2016-May (2016), pp. 1227–1234. DOI: `10.1109/APEC.2016.7468025`.

[82] Pavan Kumar Naraharisetti et al. “A linear diversity constraint - Application to scheduling in microgrids”. In: *Energy* 36.7 (2011), pp. 4235–4243. ISSN: 03605442. DOI: `10.1016/j.energy.2011.04.020`. URL: `http://dx.doi.org/10.1016/j.energy.2011.04.020`.

[83] Leo Raju, Sibi Sankar, and R. S. Milton. “Distributed optimization of solar micro-grid using multi agent reinforcement learning”. In: *Procedia Computer Science* 46.Icict 2014 (2015), pp. 231–239. ISSN: 18770509. DOI: 10.1016/j.procs.2015.02.016. URL: http://dx.doi.org/10.1016/j.procs.2015.02.016.

[84] Wencong Su, Jianhui Wang, and Jaehyung Roh. “Stochastic energy scheduling in microgrids with intermittent renewable energy resources”. In: *IEEE Transactions on Smart Grid* 5.4 (2014), pp. 1876–1883. ISSN: 19493053. DOI: 10.1109/TSG.2013.2280645.

[85] Irtaza M. Syed and Kaamran Raahemifar. “Predictive energy management and control system for PV system connected to power electric grid with periodic load shedding”. In: *Solar Energy* 136 (2016), pp. 278–287. ISSN: 0038092X. DOI: 10.1016/j.solener.2016.07.011. URL: http://dx.doi.org/10.1016/j.solener.2016.07.011.

[86] Ganesh Kumar Venayagamoorthy et al. “Dynamic Energy Management System for a Smart Microgrid”. In: *IEEE Transactions on Neural Networks and Learning Systems* 27.8 (2016), pp. 1643–1656. ISSN: 21622388. DOI: 10.1109/TNNLS.2016.2514358.

[87] Jianfang Xiao and Wang Peng. “Multiple modes control of household DC microgrid with integration of various renewable energy sources”. In: *IECON Proceedings (Industrial Electronics Conference)* (2013), pp. 1773–1778. ISSN: 1553-572X. DOI: 10.1109/IECON.2013.6699400.

[88] Yuanming Zhang and Qing Shan Jia. “Optimal storage battery scheduling for energy-efficient buildings in a microgrid”. In: *Proceedings of the 2015 27th Chinese Control and Decision Conference, CCDC 2015* (2015), pp. 5540–5545. DOI: 10.1109/CCDC.2015.7161785.

[89] Yan Zhang et al. “Optimal operation of a smart residential microgrid based on model predictive control by considering uncertainties and storage impacts”. In: *Solar Energy* 122 (2015), pp. 1052–1065. ISSN: 0038092X. DOI: 10.1016/j.solener.2015.10.027. URL: http://dx.doi.org/10.1016/j.solener.2015.10.027.

[90] Dinghuan Zhu and Gabriela Hug. “Decomposed stochastic model predictive control for optimal dispatch of storage and generation”. In: *IEEE Transactions on Smart Grid* 5.4 (2014), pp. 2044–2053. ISSN: 19493053. DOI: 10.1109/TSG.2014.2321762.

[91] Yuqing Yang et al. “Battery energy storage system size determination in renewable energy systems: A review”. In: *Renewable and Sustainable Energy Reviews* 91.January (2018), pp. 109–125. ISSN: 18790690. DOI: 10.1016/j.rser.2018.03.047. URL: https://doi.org/10.1016/j.rser.2018.03.047.

[92] Rajab Khalilpour and Anthony Vassallo. "Planning and operation scheduling of PV-battery systems: A novel methodology". In: *Renewable and Sustainable Energy Reviews* 53 (2016), pp. 194–208. ISSN: 18790690. DOI: 10.1016/j.rser.2015.08.015. URL: http://dx.doi.org/10.1016/j.rser.2015.08.015.

[93] Majid Astaneh et al. "A novel framework for optimization of size and control strategy of lithium-ion battery based off-grid renewable energy systems". In: *Energy Conversion and Management* 175.July (2018), pp. 99–111. ISSN: 01968904. DOI: 10.1016/j.enconman.2018.08.107. URL: https://doi.org/10.1016/j.enconman.2018.08.107.

[94] Carlos D. Rodríguez-Gallegos et al. "A multi-objective and robust optimization approach for sizing and placement of PV and batteries in off-grid systems fully operated by diesel generators: An Indonesian case study". In: *Energy* 160 (2018), pp. 410–429. ISSN: 03605442. DOI: 10.1016/j.energy.2018.06.185.

[95] Wouter L. Schram, Ioannis Lampropoulos, and Wilfried G.J.H.M. van Sark. "Photovoltaic systems coupled with batteries that are optimally sized for household self-consumption: Assessment of peak shaving potential". In: *Applied Energy* 223.April (2018), pp. 69–81. ISSN: 03062619. DOI: 10.1016/j.apenergy.2018.04.023. URL: https://doi.org/10.1016/j.apenergy.2018.04.023.

[96] Orlando Talent and Haiping Du. "Optimal sizing and energy scheduling of photovoltaic-battery systems under different tariff structures". In: *Renewable Energy* 129 (2018), pp. 513–526. ISSN: 18790682. DOI: 10.1016/j.renene.2018.06.016. URL: https://doi.org/10.1016/j.renene.2018.06.016.

[97] Alberto Berrueta et al. "Combined dynamic programming and region-elimination technique algorithm for optimal sizing and management of lithium-ion batteries for photovoltaic plants". In: *Applied Energy* 228.February (2018), pp. 1–11. ISSN: 03062619. DOI: 10.1016/j.apenergy.2018.06.060. URL: https://doi.org/10.1016/j.apenergy.2018.06.060.

[98] Peter Pflaum, M. Alamir, and M. Y. Lamoudi. "Battery sizing for PV power plants under regulations using randomized algorithms". In: *Renewable Energy* 113 (2017), pp. 596–607. ISSN: 18790682. DOI: 10.1016/j.renene.2017.05.091.

[99] D. L. Talavera et al. "A new approach to sizing the photovoltaic generator in self-consumption systems based on costcompetitiveness, maximizing direct self-consumption". In: *Renewable Energy* 130 (2019), pp. 1021–1035. ISSN: 18790682. DOI: 10.1016/j.renene.2018.06.088.

[100] Abid Ali et al. "Sizing and placement of battery-coupled distributed photovoltaic generations". In: *Journal of Renewable and Sustainable Energy* 9.5 (2017). ISSN: 19417012. DOI: 10.1063/1.4995531.

[101] Mohammad Reza Aghamohammadi and Hajar Abdolahinia. “A new approach for optimal sizing of battery energy storage system for primary frequency control of islanded Microgrid”. In: *International Journal of Electrical Power and Energy Systems* 54 (2014), pp. 325–333. ISSN: 01420615. DOI: 10.1016/j.ijepes.2013.07.005. URL: http://dx.doi.org/10.1016/j.ijepes.2013.07.005.

[102] Mohammad Rasol Jannesar et al. “Optimal placement, sizing, and daily charge/discharge of battery energy storage in low voltage distribution network with high photovoltaic penetration”. In: *Applied Energy* 226.March (2018), pp. 957–966. ISSN: 03062619. DOI: 10.1016/j.apenergy.2018.06.036. URL: https://doi.org/10.1016/j.apenergy.2018.06.036.

[103] Yu Ru, Jan Kleissl, and Sonia Martinez. “Storage size determination for grid-connected photovoltaic systems”. In: *IEEE Transactions on Sustainable Energy* 4.1 (2013), pp. 68–81. ISSN: 19493029. DOI: 10.1109/TSTE.2012.2199339. arXiv: 1109.4102.

[104] Johannes Weniger, Tjarko Tjaden, and Volker Quaschning. “Sizing of residential PV battery systems”. In: *Energy Procedia* 46 (2014), pp. 78–87. ISSN: 18766102. DOI: 10.1016/j.egypro.2014.01.160.

[105] Reza Hemmati and Hedayat Saboori. “Stochastic optimal battery storage sizing and scheduling in home energy management systems equipped with solar photovoltaic panels”. In: *Energy and Buildings* 152 (2017), pp. 290–300. ISSN: 03787788. DOI: 10.1016/j.enbuild.2017.07.043. URL: http://dx.doi.org/10.1016/j.enbuild.2017.07.043.

[106] Adriana C Luna et al. “Mixed-Integer-Linear-Programming-Based Energy Management System for Hybrid PV-Wind-Battery Experimental Verification”. In: *IEEE Transactions on Power Electronics* 32.4 (2017), pp. 2769–2783.

[107] James K Strayer. *Linear programming and its applications*. Springer Science & Business Media, 2012.

[108] Nocedal J Wright SJ. *Numerical optimization: Springer Science+ Business Media*. 2006.

[109] James Kennedy and Russell Eberhart. “Particle swarm optimization”. In: *Proceedings of ICNN’95-International Conference on Neural Networks*. Vol. 4. IEEE. 1995, pp. 1942–1948.

[110] Thomas H Cormen et al. *Introduction to algorithms*. MIT press, 2009.

[111] David Q Mayne et al. “Constrained model predictive control: Stability and optimality”. In: *Automatica* 36.6 (2000), pp. 789–814.

[112] Ali Mesbah. “Stochastic model predictive control: An overview and perspectives for future research”. In: *IEEE Control Systems Magazine* 36.6 (2016), pp. 30–44.

[113] Farzad Noorian and Information Technologies. “Risk Management using Model Predictive Control”. In: (2016), p. 259.

[114] Adam Hawkes and Matthew Leach. “Impacts of temporal precision in optimisation modelling of micro-combined heat and power”. In: *Energy* 30.10 (2005), pp. 1759–1779. ISSN: 03605442. DOI: 10.1016/j.energy.2004.11.012.

[115] Andrew Wright and Steven Firth. “The nature of domestic electricity-loads and effects of time averaging on statistics and on-site generation calculations”. In: *Applied Energy* 84.4 (2007), pp. 389–403. ISSN: 03062619. DOI: 10.1016/j.apenergy.2006.09.008.

[116] Sabrina Ried, Patrick Jochem, and Wolf Fichtner. “Profitability of photovoltaic battery systems considering temporal resolution”. In: (2015), pp. 5–9.

[117] L. Kools and F. Phillipson. “Data granularity and the optimal planning of distributed generation”. In: *Energy* 112 (2016), pp. 342–352. ISSN: 03605442. DOI: 10.1016/j.energy.2016.06.089. URL: http://dx.doi.org/10.1016/j.energy.2016.06.089.

[118] Jochen Linssen, Peter Stenzel, and Johannes Fleer. “Techno-economic analysis of photovoltaic battery systems and the influence of different consumer load profiles”. In: *Applied Energy* 185 (2017), pp. 2019–2025. ISSN: 03062619. DOI: 10.1016/j.apenergy.2015.11.088. URL: http://dx.doi.org/10.1016/j.apenergy.2015.11.088.

[119] Kaveh Rajab Khalilpour and Anthony Vassallo. “Technoeconomic parametric analysis of PV-battery systems”. In: *Renewable Energy* 97 (2016), pp. 757–768. ISSN: 18790682. DOI: 10.1016/j.renene.2016.06.010. URL: http://dx.doi.org/10.1016/j.renene.2016.06.010.

[120] Sylvain Quoilin et al. “Quantifying self-consumption linked to solar home battery systems: Statistical analysis and economic assessment”. In: *Applied Energy* 182 (2016), pp. 58–67. ISSN: 03062619. DOI: 10.1016/j.apenergy.2016.08.077. URL: http://dx.doi.org/10.1016/j.apenergy.2016.08.077.

[121] S. Schopfer, V. Tiefenbeck, and T. Staake. “Economic assessment of photovoltaic battery systems based on household load profiles”. In: *Applied Energy* 223.November 2017 (2018), pp. 229–248. ISSN: 03062619. DOI: 10.1016/j.apenergy.2018.03.185. URL: https://doi.org/10.1016/j.apenergy.2018.03.185.

[122] Gianfranco Chicco. “Overview and performance assessment of the clustering methods for electrical load pattern grouping”. In: *Energy* 42.1 (2012), pp. 68–80. ISSN: 03605442. DOI: 10.1016/j.energy.2011.12.031. URL: http://dx.doi.org/10.1016/j.energy.2011.12.031.

[123] Nuno Costa and Ines Matos. “Inferring daily routines from electricity meter data”. In: *Energy and Buildings* 110 (2016), pp. 294–301. ISSN: 03787788. DOI: 10.1016/j.enbuild.2015.11.015. URL: http://dx.doi.org/10.1016/j.enbuild.2015.11.015.

[124] Gianfranco Chicco and Irinel Sorin Ilie. “Support vector clustering of electrical load pattern data”. In: *IEEE Transactions on Power Systems* 24.3 (2009), pp. 1619–1628. ISSN: 08858950. DOI: 10.1109/TPWRS.2009.2023009.

[125] Bruce Stephen et al. “Enhanced load profiling for residential network customers”. In: *IEEE Transactions on Power Delivery* 29.1 (2014), pp. 88–96. ISSN: 08858977. DOI: 10.1109/TPWRD.2013.2287032.

[126] Minghao Piao and Keun Ho Ryu. “Local characterization-based load shape factor definition for electricity customer classification”. In: *IEEJ Transactions on Electrical and Electronic Engineering* 12 (2017), S110–S116. ISSN: 19314981. DOI: 10.1002/tee.22424.

[127] Joaquim L. Viegas et al. “Classification of new electricity customers based on surveys and smart metering data”. In: *Energy* 107 (2016), pp. 804–817. ISSN: 03605442. DOI: 10.1016/j.energy.2016.04.065.

[128] Y H Hsiao. “Household Electricity Demand Forecast Based on Context Information and User Daily Schedule Analysis From Meter Data”. In: *IEEE Transactions on Industrial Informatics* 11.1 (2015), pp. 33–43. DOI: 10.1109/TII.2014.2363584.

[129] Baran Yildiz et al. “Household electricity load forecasting using historical smart meter data with clustering and classification techniques”. In: *IEEE PES ISGT Asia* (2018), pp. 873–879. DOI: 10.1109/ISGT-Asia.2018.8467837.

[130] Joana M. Abreu, Francisco Câmara Pereira, and Paulo Ferrão. “Using pattern recognition to identify habitual behavior in residential electricity consumption”. In: *Energy and Buildings* 49 (2012), pp. 479–487. ISSN: 03787788. DOI: 10.1016/j.enbuild.2012.02.044. URL: http://dx.doi.org/10.1016/j.enbuild.2012.02.044.

[131] Jungsuk Kwac, June Flora, and Ram Rajagopal. “Household energy consumption segmentation using hourly data”. In: *IEEE Transactions on Smart Grid* 5.1 (2014), pp. 420–430. ISSN: 19493053. DOI: 10.1109/TSG.2013.2278477.

[132] George J. Tsekouras, Nikos D. Hatziargyriou, and Evangelos N. Dialynas. “Two-stage pattern recognition of load curves for classification of electricity customers”. In: *IEEE Transactions on Power Systems* 22.3 (2007), pp. 1120–1128. ISSN: 08858950. DOI: 10.1109/TPWRS.2007.901287.

[133] Gianfranco Chicco, Roberto Napoli, and Federico Piglione. “Comparisons among clustering techniques for electricity customer classification”. In: *IEEE Transactions on Power Systems* 21.2 (2006), pp. 933–940. ISSN: 08858950. DOI: 10.1109/TPWRS.2006.873122.

[134] Fintan McLoughlin, Aidan Duffy, and Michael Conlon. "A clustering approach to domestic electricity load profile characterisation using smart metering data". In: *Applied Energy* 141 (2015), pp. 190–199. ISSN: 03062619. DOI: 10.1016/j.apenergy.2014.12.039. URL: http://dx.doi.org/10.1016/j.apenergy.2014.12.039.

[135] David Gerbec et al. "Allocation of the load profiles to consumers using probabilistic neural networks". In: *IEEE Transactions on Power Systems* 20.2 (2005), pp. 548–555. ISSN: 08858950. DOI: 10.1109/TPWRS.2005.846236.

[136] Florentin Batrinu et al. "Comparisons Among Clustering Techniques for Electricity Customer Classificatio". In: 21.2 (2006), pp. 1–7. DOI: 10.1109/TPWRS.2006.873122.

[137] Omid Motlagh et al. "Analysis of household electricity consumption behaviours: Impact of domestic electricity generation". In: *Applied Mathematics and Computation* 270 (2015), pp. 165–178. ISSN: 00963003. DOI: 10.1016/j.amc.2015.08.029. URL: http://dx.doi.org/10.1016/j.amc.2015.08.029.

[138] Anthony R. Florita et al. "Classification of Commercial Building Electrical Demand Profiles for Energy Storage Applications". In: *Journal of Solar Energy Engineering* 135.3 (2013), p. 031020. ISSN: 0199-6231. DOI: 10.1115/1.4024029. URL: http://solarenergyengineering.asmedigitalcollection.asme.org/article.aspx?doi=10.1115/1.4024029.

[139] Mahmoud Ghofrani et al. "A framework for optimal placement of energy storage units within a power system with high wind penetration". In: *IEEE Transactions on Sustainable Energy* 4.2 (2013), pp. 434–442. ISSN: 19493029. DOI: 10.1109/TSTE.2012.2227343.

[140] Yongxi Zhang et al. "Optimal allocation of battery energy storage systems in distribution networks with high wind power penetration". In: *IET Renewable Power Generation* 10.8 (2016), pp. 1105–1113. ISSN: 1752-1416. DOI: 10.1049/iet-rpg.2015.0542.

[141] Bernd Klöckl and George Papaefthymiou. "Multivariate time series models for studies on stochastic generators in power systems". In: *Electric Power Systems Research* 80.3 (2010), pp. 265–276. ISSN: 03787796. DOI: 10.1016/j.epsr.2009.09.009.

[142] K. Suomalainen et al. "Synthetic wind speed scenarios including diurnal effects: Implications for wind power dimensioning". In: *Energy* 37.1 (2012), pp. 41–50. ISSN: 03605442. DOI: 10.1016/j.energy.2011.08.001. URL: http://dx.doi.org/10.1016/j.energy.2011.08.001.

[143] J. Chen, J.S. Kim, and C. Rabiti. "Probabilistic analysis of hybrid energy systems using synthetic renewable and load data". In: *Proceedings of the American Control Conference* (2017), pp. 4723–4728. ISSN: 07431619. DOI: 10.23919/ACC.2017.7963685.

[144] Yuchen Tang et al. “Evaluating the variability of photovoltaics: A new stochastic method to generate site-specific synthetic solar data and applications to system studies”. In: *Renewable Energy* 133 (2019), pp. 1099–1107. ISSN: 18790682. DOI: 10.1016/j.renene.2018.10.102. URL: https://doi.org/10.1016/j.renene.2018.10.102.

[145] Tao Wang, Hsiao-dong Chiang, and Ryuya Tanabe. “Toward a Flexible Scenario Generation Tool for Stochastic Renewable Energy Analysis”. In: *Power Systems Computation Conference (PSCC) 2016* (2016).

[146] Emil B. Iversen, Pierre Pinson, and Igor Arduin. “RESGen: Renewable Energy Scenario Generation Platform”. In: *2016 IEEE Power & Energy Society General Meeting (PESGM)* (2016), pp. 1–5.

[147] A. Grandjean, J. Adnot, and G. Binet. “A review and an analysis of the residential electric load curve models”. In: *Renewable and Sustainable Energy Reviews* 16.9 (2012), pp. 6539–6565. ISSN: 13640321. DOI: 10.1016/j.rser.2012.08.013. URL: http://dx.doi.org/10.1016/j.rser.2012.08.013.

[148] Lukas G. Swan and V. Ismet Ugursal. “Modeling of end-use energy consumption in the residential sector: A review of modeling techniques”. In: *Renewable and Sustainable Energy Reviews* 13.8 (2009), pp. 1819–1835. ISSN: 13640321. DOI: 10.1016/j.rser.2008.09.033.

[149] Michael Parti and C Parti. “The Total and Appliance-Specific Conditional Demand for Electricity in the Household Sector”. In: *The Bell Journal of Economics* 11.1 (1980), pp. 309–321.

[150] Cheng Hsiao, Dean C. Mountain, and Kathleen Ho Sllwian. “A bayesian integration of end-use metering and conditional-demand analysis”. In: *Journal of Business and Economic Statistics* 13.3 (1995), pp. 315–326. ISSN: 15372707. DOI: 10.1080/07350015.1995.10524605.

[151] Merih Aydinalp-Koksal and V. Ismet Ugursal. “Comparison of neural network, conditional demand analysis, and engineering approaches for modeling end-use energy consumption in the residential sector”. In: *Applied Energy* 85.4 (2008), pp. 271–296. ISSN: 03062619. DOI: 10.1016/j.apenergy.2006.09.012.

[152] G. Mihalakakou, M. Santamouris, and A. Tsangrassoulis. “On the energy consumption in residential buildings”. In: *Energy and Buildings* 34.7 (2002), pp. 727–736. ISSN: 03787788. DOI: 10.1016/S0378-7788(01)00137-2.

[153] A. Capasso et al. “A bottom-up approach to residential load modeling”. In: *IEEE Transactions on Power Systems* 9.2 (1994), pp. 957–964. ISSN: 15580679. DOI: 10.1109/59.317650.

[154] Ipsos-RSL Office for National Statistics. *United Kingdom Time Use Survey, 2000.* https://beta.ukdataservice.ac.uk/datacatalogue/studies/study?id=4504. [Online; accessed 26-September-2019]. 2003. DOI: 10.5255/UKDA-SN-4504-1.

[155] Ian Goodfellow et al. “Generative adversarial nets”. In: *Advances in neural information processing systems*. 2014, pp. 2672–2680.

[156] Alec Radford, Luke Metz, and Soumith Chintala. “Unsupervised Representation Learning With Deep Convolutional Generative Adversarial Networks”. In: *arXiv preprint arXiv:1511.06434* (2016), pp. 1–16. ISSN: 0004-6361. DOI: `10.1051/0004-6361/201527329`. arXiv: `1511.06434`. URL: `https://arxiv.org/pdf/1511.06434.pdf`.

[157] Yuxuan Gu et al. “Gan-based model for residential load generation considering typical consumption patterns”. In: *2019 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*. IEEE. 2019, pp. 1–5.

[158] Rui Tang et al. “Impacts of Temporal Resolution and System Efficiency on PV Battery System Optimisation”. In: *2017 Asia-Pacific Solar Research Conference* (2017). URL: `http://apvi.org.au/solar-research-conference/wp-content/uploads/2017/12/029_R-Tang_DI_Paper_Peer-reviewed.pdf`.

[159] Lucas Theis, Aäron van den Oord, and Matthias Bethge. “A note on the evaluation of generative models”. In: *arXiv preprint arXiv:1511.01844* (2015).

[160] Solar Analytics. *Connect With Your Solar*. `https://www.solaranalytics.com/au/`. [Online; accessed 10-October-2019]. 2019.

[161] Wattwatchers. *Wattwatchers: Super-smart devices for energy monitoring*. `https://wattwatchers.com.au/`. [Online; accessed 02-November-2018]. 2018.

[162] LLC Gurobi Optimization. *Gurobi Optimizer Reference Manual*. 2020. URL: `http://www.gurobi.com`.

[163] Rui Tang, Jonathon Dore, and John Laird. “Site Specific Battery Simulation Model”. In: ().

[164] Australian Energy Council (AEC). *Renewable Energy in Australia - how do we really compare?* `https://www.energycouncil.com.au/media/1318/2016-06-23_aec-renewables-factsheet.pdf`. [Online; accessed 01-June-2018]. 2016.

[165] Australian PV Institute (APVI). *Australian PV Institute (APVI) Solar Map, funded by the Australian Renewable Energy Agency*. `https://pv-map.apvi.org.au`. [Online; accessed 25-April -2018]. 2018.

[166] A. Jäger-Waldau. *PV Status Report 2017*. `http://publications.jrc.ec.europa.eu/repository/bitstream/JRC108105/kjna28817enn.pdf`. [Online; accessed 01-July-2018]. 2017.

[167] Stephen Haben, Colin Singleton, and Peter Grindrod. “Analysis and clustering of residential customers energy behavioral demand using smart meter data”. In: *IEEE Transactions on Smart Grid* 7.1 (2016), pp. 136–144. ISSN: 19493053. DOI: `10.1109/TSG.2015.2409786`.

[168] Joshua D. Rhodes et al. "Clustering analysis of residential electricity demand profiles". In: *Applied Energy* 135 (2014), pp. 461–471. ISSN: 03062619. DOI: 10.1016/j.apenergy.2014.08.111. URL: http://dx.doi.org/10.1016/j.apenergy.2014.08.111.

[169] H. Hino et al. "A Versatile Clustering Method for Electricity Consumption Pattern Analysis in Households". In: *IEEE Transactions on Smart Grid* 4.2 (2013), pp. 1048–1057. ISSN: 1949-3053. DOI: 10.1109/TSG.2013.2240319. URL: http://ieeexplore.ieee.org/ielx7/5165411/6517533/06484217.pdf?tp={\&}arnumber=6484217{\&}isnumber=6517533{\%}5Cnhttp://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=6484217.

[170] Bureau of Meteorology. *Renewable Energy in Australia - how do we really compare?* http://www.bom.gov.au/climate/glossary/seasons.shtml. [Online; accessed 19-June-2018]. no date.

[171] J. MacQueen. "Some methods for classification and analysis of multivariate observations". In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics.* Berkeley, Calif.: University of California Press, 1967, pp. 281–297. URL: https://projecteuclid.org/euclid.bsmsp/1200512992.

[172] Leon Bottou and Yoshua Bengio. "Convergence properties of the k-means algorithms". In: *Advances in neural information processing systems.* 1995, pp. 585–592.

[173] David L Davies and Donald W Bouldin. "A cluster separation measure". In: *IEEE transactions on pattern analysis and machine intelligence* PAMI-1.2 (1979), pp. 224–227.

[174] Theodore Wilbur Anderson et al. *An introduction to multivariate statistical analysis.* Vol. 2. Wiley New York, 1958.

[175] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[176] Tin Kam Ho. "Random decision forests". In: *Document analysis and recognition, 1995., proceedings of the third international conference on.* Vol. 1. IEEE. 1995, pp. 278–282.

[177] Leo Breiman. "Bagging predictors". In: *Machine learning* 24.2 (1996), pp. 123–140.

[178] Aurélien Géron. *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems.* " O'Reilly Media, Inc.", 2017.

[179] Robert Tibshirani. "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1996), pp. 267–288.

[180] Miron B Kursa and Witold R Rudnicki. "Feature selection with the Boruta package". In: *J Stat Softw* 36.11 (2010), pp. 1–13.

[181] Frauke Degenhardt, Stephan Seifert, and Silke Szymczak. "Evaluation of variable selection methods for random forests and omics data sets". In: *Briefings in Bioinformatics* (2017), bbx124. DOI: 10.1093/bib/bbx124. eprint: /oup/backfile/content_public/journal/bib/pap/10.1093_bib_bbx124/1/bbx124.pdf. URL: http://dx.doi.org/10.1093/bib/bbx124.

[182] H Daniel. *boruta py*. https://github.com/scikit-learn-contrib/boruta_py. 2016.

[183] James Bergstra and Yoshua Bengio. "Random Search for Hyper-parameter Optimization". In: *J. Mach. Learn. Res.* 13 (Feb. 2012), pp. 281–305. ISSN: 1532-4435. URL: http://dl.acm.org/citation.cfm?id=2188385.2188395.

[184] Solar Choice. *Is home solar battery storage worth it? (Jan 2018 update)*. https://www.solarchoice.net.au/blog/home-solar-battery-storage-worth-it-2018. [Online; accessed 29-August-2018]. 2018.

[185] Solar Quotes. *Solar Battery Storage Comparison Table*. https://www.solarquotes.com.au/battery-storage/comparison-table/. [Online; accessed 16-September-2018]. 2018.

[186] Ian Goodfellow. "NIPS 2016 Tutorial: Generative Adversarial Networks". In: *arXiv preprint arXiv:1701.00160* (2016). ISSN: 0253-0465. DOI: 10.1001/jamainternmed.2016.8245. arXiv: 1701.00160. URL: http://arxiv.org/abs/1701.00160.

[187] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.

[188] Steve Lawrence et al. "Face recognition: A convolutional neural-network approach". In: *IEEE transactions on neural networks* 8.1 (1997), pp. 98–113.

[189] Augustus Odena, Vincent Dumoulin, and Chris Olah. "Deconvolution and Checkerboard Artifacts". In: *Distill* (2016). DOI: 10.23915/distill.00003. URL: http://distill.pub/2016/deconv-checkerboard.

[190] Sergey Ioffe and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *arXiv preprint arXiv:1502.03167* (2015).

[191] Vinod Nair and Geoffrey E Hinton. "Rectified linear units improve restricted boltzmann machines". In: *Proceedings of the 27th international conference on machine learning (ICML-10)*. 2010, pp. 807–814.

[192] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. "Rectifier nonlinearities improve neural network acoustic models". In: *Proc. icml*. Vol. 30. 2013, p. 3.

[193] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[194] Nitish Srivastava et al. "Dropout: a simple way to prevent neural networks from overfitting". In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[195] Mehdi Mirza and Simon Osindero. "Conditional generative adversarial nets". In: *arXiv preprint arXiv:1411.1784* (2014).

[196] Emily L Denton, Soumith Chintala, Rob Fergus, et al. "Deep generative image models using aLaplacian pyramid of adversarial networks". In: *Advances in neural information processing systems*. 2015, pp. 1486–1494.

[197] Yarin Gal and Zoubin Ghahramani. "A theoretically grounded application of dropout in recurrent neural networks". In: *Advances in neural information processing systems*. 2016, pp. 1019–1027.

[198] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. "Which training methods for GANs do actually converge?" In: *arXiv preprint arXiv:1801.04406* (2018).

[199] Roger B Nelsen. *An introduction to copulas*. Springer Science & Business Media, 2007.

[200] Marius Hofert et al. *copula: Multivariate Dependence with Copulas*. R package version 0.999-19.1. 2018. URL: `https://CRAN.R-project.org/package=copula`.

[201] Solar Choice. *Solar Choice Solar & Battery Storage Sizing & Payback Calculator*. `https://www.solarchoice.net.au/blog/solar-pv-battery-storage-sizing-payback-calculator`. [Online; accessed 29-August-2019]. 2019.

[202] Khalid Abdulla et al. "The importance of temporal resolution in evaluating residential energy storage". In: *Power & Energy Society General Meeting, 2017 IEEE*. IEEE. 2017, pp. 1–5.

[203] Goldie-Scot, Logan. *A Behind the Scenes Take on Lithium-ion Battery Prices*. `https://about.bnef.com/blog/behind-scenes-take-lithium-ion-battery-prices/`. [Online; accessed 29-December-2019]. 2019.

[204] Rui Tang et al. "Residential battery sizing model using net meter energy data clustering". In: *Applied Energy* 251 (2019), p. 113324. DOI: `10.1016/j.apenergy.2019.113324`.

[205] Martín Abadi et al. "Tensorflow: A system for large-scale machine learning". In: *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*. 2016, pp. 265–283.

[206] François Chollet et al. *Keras*. `https://keras.io`. 2015.

[207] Jianhua Lin. "Divergence measures based on the Shannon entropy". In: *IEEE Transactions on Information theory* 37.1 (1991), pp. 145–151.

[208] J. Polo et al. “A simple approach to the synthetic generation of solar irradiance time series with high temporal resolution”. In: *Solar Energy* 85.5 (2011), pp. 1164–1170. ISSN: 0038092X. DOI: 10.1016/j.solener.2011.03.011. URL: http://dx.doi.org/10.1016/j.solener.2011.03.011.

[209] M. Larrañeta et al. “An improved model for the synthetic generation of high temporal resolution direct normal irradiation time series”. In: *Solar Energy* 122 (2015), pp. 517–528. ISSN: 0038092X. DOI: 10.1016/j.solener.2015.09.030.

[210] A. P. Grantham et al. “Generating synthetic five-minute solar irradiance values from hourly observations”. In: *Solar Energy* 147 (2017), pp. 209–221. ISSN: 0038092X. DOI: 10.1016/j.solener.2017.03.026. URL: http://dx.doi.org/10.1016/j.solener.2017.03.026.

[211] Guolong Liu et al. “Super Resolution Perception for Smart Meter Data”. In: *Information Sciences* 526 (2020), pp. 263–273. ISSN: 00200255. DOI: 10.1016/j.ins.2020.03.088.

[212] Christian Ledig et al. “Photo-realistic single image super-resolution using a generative adversarial network”. In: *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017* 2017-Janua (2017), pp. 105–114. ISSN: 1063-6919. DOI: 10.1109/CVPR.2017.19. arXiv: 1609.04802.

[213] Xiaojiao Mao, Chunhua Shen, and Yu-Bin Yang. “Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections”. In: *Advances in neural information processing systems*. 2016, pp. 2802–2810.

[214] Ossama Abdel-Hamid et al. “Convolutional neural networks for speech recognition”. In: *IEEE/ACM Transactions on audio, speech, and language processing* 22.10 (2014), pp. 1533–1545.

[215] Andrej Karpathy and Li Fei-Fei. “Deep visual-semantic alignments for generating image descriptions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3128–3137.

[216] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[217] Kaiming He et al. “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1026–1034.

[218] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. “Learning representations by back-propagating errors”. In: *nature* 323.6088 (1986), pp. 533–536.

[219] RF Woolson. “Wilcoxon signed-rank test”. In: *Wiley encyclopedia of clinical trials* (2007), pp. 1–3.