Development of a Practical Controller for the Optimised Control of Hot-Water in Grid-Connected Microgrids and Embedded Networks

Matthew Grant

2024

AUT School of Creative Technologies, Department of Electrical and Electronic Engineering

A thesis submitted to

Auckland University of Technology

in fulfilment of the requirements for the degree of

Master of Engineering

You are the men and women who have to do the work, and I say again, the great transformation that is necessary is in your hands.

Michael Joseph Savage [1]

Abstract

As New Zealand's electricity industry braces for disruptive change in the face of increasing public consciousness to the worsening impacts of climate change, evolving government policy and the impositions of electrification to transport and industrial heating, the question of managing the country's energy infrastructure is emerging as a critical component in the development of a more flexible electrical grid.

This thesis examines the role of hot water load control and its potential to contribute to the problem of managing New Zealand's peak energy demands. Drawing from historical events such as the localised power cuts of August 2021, this study explores the effectiveness of traditional hot water ripple control and the centralised load control of hot water cylinders. While serving New Zealand well, the effectiveness of traditional ripple control (RC) is limited in its current state, requiring an exploration of more modern, flexible approaches to demand response, especially in the face of evolving technologies such as electric vehicles (EVs) and residential heat pumps.

By developing and validating a digital model of a hot water cylinder and exploring computer algorithms for its smart control in the context of a grid connected microgrid, this research estimates the potential for hot water cylinders to satisfy end-user needs centred around comfort and cost while also benefiting the wider electricity grid by reducing peak electricity demand.

Potential approaches to hot water load control including dynamic programming, shortest path search algorithms, and reinforcement learning are investigated as strategies to automate and optimize hot water load control, offering insights into the potential contribution the improved management of the humble residential hot water cylinder could provide to New Zealand on its journey towards a more sustainable, flexible electricity system.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AI	Artificial Intelligence
ACO	Ant Colony Optimisation
ADC	Analogue to Digital Conversion
ANN	Artificial Neural Network
BESS	Battery Energy Storage System
BRANZ	Building Research Association of New Zealand
CFD	Computational Fluid Dynamics
CFL	Compact Fluorescent Lamp
CJC	Cold Junction Compensation
DAG	Directed Acyclic Graph
DAQ	Data Acquisition
DER	Distributed Energy Resource
DHW	Domestic Hot Water
DNO	Distribution Network Operator
DR	Demand Response
DSM	Demand Side Management
DP	Dynamic Programming
EA	Electricity Authority
EECA	Energy Efficiency and Conservation Authority
EDB	Electricity Distribution Business
EV	Electric Vehicle
EWH	Electric Water Heater
GA	Genetic Algorithm
GB	Gigabyte
GW	Gigawatt
HEEP	Household Energy End-use Project
HVAC	Heating, Ventilation and Air Conditioning
IANZ	International Accreditation New Zealand
ICCC	Interim Climate Change Commission
ICP	Installation Control Point

IPAG	Industry Participation and Advisory Group
IVP	Initial Value Problem
kW	Kilowatt
kWh	Kilowatt hour
LED	Light Emitting Diode
LP	Linear Programming
LV	Low Voltage
MATLAB	Matrix Laboratory
MB	Moving Boundary
MBIE	Ministry of Business, Innovation and Employment
MEPS	Minimum Energy Performance Standard
MILP	Mixed Integer Linear Programming
MIT	Massachusetts Institute of Technology
ML	Machine Learning
MPC	Model Predictive Control
MSTL	Multiple Seasonality Decomposition using LOESS
NILM	Non-intrusive Load Monitoring
NIWA	National Institute of Water and Atmospheric Research
NSP	Network Supply Point
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
PF	Plug Flow
PSO	Particle Swarm Optimisation
PV	Photovoltaic
REDD	Reference Energy Disaggregation Dataset
RL	Reinforcement Learning
RMSE	Root Mean Square Error
SOC	State of Charge
TES	Thermal Energy Storage
TPR	Temperature/Pressure Relief

Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

_____ 26/09/2024

Signature Date

Acknowledgements

This thesis is dedicated to the memory of my late academic supervisor, Dr. Craig Baguley, who devoted his life to the engineering profession and the students he taught.

Craig was humble, hardworking and completely genuine. His passing was a tragedy not just for the staff and students of AUT University, but for the entire New Zealand engineering community.

There are many others I would also like to thank, without whose help and understanding this project would never have been completed.

Dr. Jeff Kilby, in particular, who stepped in after the passing of Craig, deserves my heartfelt thanks. His encouragement and positivity were the driving force in seeing this thesis through to completion. Jeff is another of those dedicated to his students, who works tirelessly every day to give them the best chance of success.

The laboratory technicians at AUT University have always impressed me with their knowledge and expertise. I would especially like to thank Stephen Hartley for his help in setting up the hot water cylinder. Without his input, this part of the research would not have been possible.

I am also grateful to Shay Brazier of Re/volve Energy, who's vision for an electrified future provided the original ideas for this research. He continues to inspire me every day through his work in making that vision reality.

To my late father, who instilled in me his sense of curiosity, and to my mother who gave me a love of books. I will always be grateful for their love and the many sacrifices they made along the way.

Finally, I would like to thank my wife Ayako and young boys Cian and Tomi. Your unwavering support and love have been a source of strength on this journey. Everything else from now on, I devote entirely to you.

Chapter 1 Introduction

New Zealand's electricity industry has now started on a period of disruptive change [2] as it navigates through shifts in government policy and heightened public awareness concerning the impending effects of climate change. Improved capabilities in the technologies associated with renewable energy generation and storage have changed the public's perceptions of what can be done to prevent further harm to global ecosystems. Similarly, changes in government policy, including targets mandated by the Climate Change Response (Zero-Carbon) Amendment Act 2019, have created a political imperative to make the goal of 'rapid electrification' [3] a reality. New Zealand is now faced with the question of how to implement far-reaching changes to its underlying energy infrastructure in a very short time, as there is a real possibility that the cumulative effects of the approaching change will be felt 'sooner then we think' [4].

1.1 Managing the Peak

Already, there are worrying signs that there is very little room to move and that the natural limits of the current system may be uncomfortably close.

On 9 August 2021, electricity demand in the country reached 'an all-time high' [5] as New Zealanders huddled around their heaters at home on a particularly cold winter evening. The resulting blackout left 34,000 homes without power for more than two hours [6]. Subsequent weather related grid warnings, most recently caused by a cold snap coinciding with low wind forecasts and reduced power generation capacity due to plant maintenance [7], leave questions around the fragility of the national electrical system is its ability to cope with the unpredictable world that approaches.

Transpower's influential report, Whakamana I Te Mauri Hiko, has estimated that the New Zealand government may need to add as much as 68% more generation to keep up with the projected 40% increase in demand from 7.3 GW to 10 GW by 2050 [8].

Given the financial outlay this will entail ($14-24 billion spent on new capacity through to 2050) [8], we may as a small country face an unnecessarily burdensome cost should we choose the 'business as usual' approach of overbuilding new generation capacity to match new demand [4].

1.1.1 Demand Side Management (DSM) and Demand Response (DR)

One alternative to matching supply capacity to demand is to invert the traditional paradigm and change electricity demand to match generation. First popularised in the United States in the 1970s [9], the concept of Demand Side Management (DSM) seeks to flatten load profiles using techniques such as peak demand clipping, valley filling, and load shifting by influencing consumer behaviour through offering incentives [9]. Demand Response (DR) is achieved when end users 'intentionally' change their behaviour to affect either instantaneous demand or energy consumption in response to signals from electricity suppliers, which may include price signals or incentive payments [10].

For instance, it is estimated that each gigawatt of avoided peak demand in the transmission and distribution system could save $1.5 billion [8]. Similarly, a study in [11] estimates savings of $0.28 billion by 2030 and $2.6 billion by 2050 if measures are taken to reduce transmission capacity.

It must be stated clearly, however, that DSM and DR measures do not reduce the overall energy flowing in the network or its ability to meet the country's energy needs [12]. Since demand response is aimed at influencing 'electricity consumption patterns', the load is instead shifted out of peak demand periods, reducing the need for expensive peak generation [12]. As well as allowing for the deferment of generation capacity, successful demand response technologies can also enable the deferment of distribution network investment – possibly the highest value demand response service [13].

1.1.2 Ripple Control

New Zealand has invested heavily in DR and enjoyed some success with the centralised control of hot water known as ripple control. Ripple control is a form of direct load control of hot water cylinders pioneered during the 1950s [14]. The system detects signals embedded within the mains electricity waveform, which activate or deactivate relays connected to the cylinder's heating element. The system is relatively cheap and effective [14]. It has served the country well by providing Electricity Distribution Businesses (EDB) with the ability to quickly curtail the water heating component of their load profile at critical moments to shift peak demand load [15].

Unfortunately, the technology is ageing and provides for blunt control of a dwindling number of hot water cylinders [16]. The system is also no longer coordinated by a central authority and has become so fragmented that there is no longer a clear knowledge of how large the controllable resource is [17]. One estimate puts the value of ripple control to the country at around 734 MW [18], whereas [14] estimates the load connected to ripple control systems represents 15% of New Zealand's peak demand. Whatever the true number, the ripple control system still sees significant investment and maintenance in various parts of the country and it will likely remain an important part of the power system [14]. In the longer term, however, there is a consensus that new technologies that will offer more efficient and 'nuanced' forms of demand response around hot water will be developed, especially as the country moves into a new era in which electric vehicles (EV) and heat pumps become more commonplace [11].

1.1.3 Modern Hot Water Control

This thesis explores what an alternative solution might look like and its effects on a local network. A central contention of the research presented in this thesis is that the full potential of the electric hot water cylinder has yet to be realised, and there could be value in repurposing a simple but reliable energy storage device to meet the demands of the highly distributed and technologically focussed smart grid of the future.

Ripple control, as revolutionary as it was when first developed, is different from the fine grain control required of modern DER capable electricity systems. It is suited to sporadic use on large blocks of customers, whose individual heating needs are invisible to the local EDB and system operator.

This thesis will make the case that equipping hot water cylinders with the ability to manage themselves according to signals supplied by the market, the system operator and the comfort needs of the end user has the potential to offer significant benefits to the wider electricity system, save EDB's money through deferring the need to strengthen local networks while saving residential consumers money by heating water at strategic times of the day when the cost to do so is lowest.

1.2 Research Methodology

The approach taken in this research will:

1. Investigate common energy demand profiles as measured in New Zealand households based on data acquired from the GREEN Grid project.
2. Estimate individual household hot water draw profiles from historical household hot water cylinder data captured from GREEN Grid project households.
3. Develop a generic digital model of a hot water cylinder in MATLAB and Simulink.
4. Validate the hot water cylinder model with reference to experimental hot water cylinder measurements in a laboratory setting.
5. Investigate several approaches to the smart control of hot water cylinder heating using computer algorithms coded in MATLAB.
6. Demonstrate the benefits to a local electricity network for a community of houses equipped with smart hot water cylinders and attempt to quantify this effect in terms of energy savings and reductions in peak electricity demand.

1.3 Thesis Structure

The structure of the thesis is broken down into the following chapters:

Chapter 1 introduces the role played by the residential hot water cylinder in the context of the challenges faced by the wider electrical grid in New Zealand today and describes the methodology on which this study is based;

Chapter 2 discusses the various sources of information on residential hot water control that were consulted and the findings of an extensive literature review;

Chapter 3 introduces data from the GREEN Grid project and describe trends seen in the data that help characterize residential load profiles in New Zealand;

Chapter 4 describes the steps to test a hot water cylinder in a laboratory setting. The results of this testing are used to validate the models built in MATLAB/Simulink[1];

Chapter 5 describes common mathematical modelling approaches used to simulate the internal behaviour of hot water storage devices. Modelling from the simplest single body formulations, to more complex systems of multiple ordinary differential equations are presented;

Chapter 6 will use heating data collected in the laboratory and described in chapter 4, to validate hot water cylinders models of varying complexity against real world experimental data;

Chapter 7 will introduce the idea of Dynamic Programming (DP) and variations of it, including shortest path graph search techniques using Dijkstra's algorithm and A* search to automate the challenge of scheduling hot water demand to avoid peak demand congestion;

Chapter 8 will simulate aspects of the embedded network and associated community introduced in Chapter 3 to see whether the effects of hot water control are meaningful enough to ameliorate the increased demands on the grid expected after large scale EV adoption and widespread photovoltaic (PV) generation;

Chapter 9 will suggest a basic approach that might be taken to develop a hot water controller for a single household in New Zealand and the ways this might be generalised for the benefit of a small community around an embedded network; and

Chapter 10 will conclude with a summary of the results and suggestions for further work in this area.

[1] MATLAB/Simulink, (an abbreviation of "MATrix LABoratory" is a proprietary multi-paradigm programming language and numeric computing environment developed by MathWorks [19].

Chapter 2 Literature Survey

2.1 Literature Survey

The first use of electricity to heat water in residential buildings in New Zealand occurred when the first electrified immersion water heaters were introduced in the early 1920s [20]. By the 1930s, the technology around electrified residential cylinders had matured to a form recognisable today, and as described by Williamson and Sue in their study of domestic hot water use in New Zealand [20], had become the dominant means of residential hot water heating as early as the 1950s.

Today, the residential household sector in New Zealand is responsible for approximately one-third of the country's electricity demand [21], and by some estimates, residential hot water heating consumes as much as 12 per cent of the country's total electricity use [22].

Such a significant contribution to the national electrical load has received relatively little attention in the commercial and academic worlds, partly because of the effectiveness of ripple control over the years to help manage peak demand and grid stability at times of extreme stress. As explained in a summary prepared by the Energy Efficiency Conservation Authority (EECA) [14], ripple control is still an important tool in managing the New Zealand electrical grid. Its influence will continue to be felt well into the future.

However, recent improvements in telecommunications and microprocessor technologies and the enormous increase in forecast demand due to 'accelerated electrification' represent a challenge to the status quo which may see large-scale hot water control become the subject of renewed interest [23].

Though much work must be done to understand its true potential in a modern electricity grid, more advanced forms of distributed hot water control will become increasingly important to balance the effects of increasingly distributed forms of renewable generation.

This section will describe the results of a systematic literature survey, which will inform the development of the key research questions to form the basis of this thesis.

2.2 Systematic Literature Review

A systematic literature review was undertaken at the beginning of the research to probe recent developments in hot water load control methods in the international and local energy industries. The literature review was not limited to academic sources; it also sought examples of hot water load control in the commercial market and the New Zealand electricity industry.

2.2.1 Academic Literature

Three well-known engineering databases, IEEE Xplore, Science Direct and Web of Science, were surveyed for relevant peer-reviewed journal articles between 2010 and 2021.

The survey was conducted systematically using the keyword search:

(”hot water” OR “hot water cylinder” OR “domestic hot water” OR “residential hot water” OR “electric water heater” OR EWH OR “hot water consumption” OR “thermostatic load” OR “thermostatic* controlled load” OR DHW) AND (control OR controller*) AND (“demand response” OR “DR” OR “demand side management” OR “DSM”)

Articles were limited to peer-reviewed journals published between 2010 and 2021.

Use of the IEEE Xplore database identified 2145 items, of which 277 were journals published within the relevant timeframe. According to the prescribed search criteria, sorting eliminated 233 titles, with 44 remaining. The same search of Web of Science yielded 645 results, of which 565 were eliminated and 51 retained.

2.2.2 Characteristics of Hierarchical and Distributed Control

Many traditional hot water control schemes are designed with a centralised control topology, where decisions to heat aggregated units of water storage devices are made through one or two way communication from a centralised control source [24], such as is the case for ripple control.

There are however alternatives to the centralised control architecture, including hierarchical and distributed forms of control [24].

Distributed schemes are most associated with DR and DSM approaches and represent a more customer focussed approach to user comfort and a utility in a modern network, since control of the state of the hot water storage device remains with the user.

Hierarchical arrangements on the other hand involve a two-step control process. A local controller acts as an intermediary who remains in communication with aggregations of electric water heaters, keeping track of their respective states. The local controller might pass on a control signal received from the system operator to the individual units within its jurisdiction [24]. In this scheme as well, the final decision of whether to participate would be subject to a control policy decided by the hot water user, depending on the state of the hot water cylinder (temperature) at the time the control signal was sent and received.

Ideally, hot water control strategies operating within hierarchical or distributed networks are not subject solely to the rigid direction of the system operator. A perfect residential hot water cylinder controller should also operate according to the needs of the consumers it serves. This type of operation is naturally more 'customer centred' [25], as the system takes in information for sensors and environmental cues to decide on the best course of action, whether to heat the water in the hot water cylinder or not.

2.2.3 Heating Strategies

Various heating strategies are available that range in complexity from simple thermostatic control to advanced machine learning using Artificial Neural Networks (ANN).

A useful distinction in categorising the varied control approaches is made by Brence et al. [26], where 'static' approaches refer to simple 'rule-based' control in response to sensor data such as temperature, flow rate or tank water level.

Table 1. Simple heating control strategies.

Method	**Description**
Thermostat control	Thermostat turns on and off around hysteresis limit. Most common method for typical hot water cylinders. (Bang-bang control)
Timer control	Rudimentary scheduling
Load control	Solid state relay/ thermistor (ripple control/pilot wire)
Manual on/off	Extended periods of interrupted usage/absence

More advanced techniques, especially those involving Artificial Intelligence (AI), use "oracle" approaches [26]. An oracle is a prediction algorithm, often employing machine learning of historical user data to estimate the likely water demand profile for a given day based on historical behaviour patterns.

Oracle methods use prediction algorithms to estimate consumer water demand to find optimal heating schedules. There are several approaches to determining the optimal heating schedule required for an expected schedule of hot water demand. The five major categories encountered in the survey are summarised in Table 2.

Table 2. Intelligent approach categories.

Approach	Characteristics	Examples in survey
Linear Programming (LP) / Mixed Integer Linear Programming (MILP)	Model free Computationally complex External solver	[27] [28] [29]
Dynamic Programming (DP)	Model based	[30] [31]
Model Predictive Control (MPC)	Real time control/ Model based/ computationally complex	[32] [33] [34] [35] [36] [37] [38] [39]
Artificial Intelligence (AI) optimizations: • Genetic Algorithm (GA) • Particle Swarm Optimisation (PSO) • Ant Colony Optimisation (ACO) etc. • Fuzzy logic	Model free	[40] [41] [42]
Reinforcement Learning (RL)	Model based Model free	[43], [44] [45] [46] [47] [48] [46] [49] [50]

AI is a broad identifier spanning computer science and statistical computing techniques ranging from classical dynamic programming to modern deep learning with ANNs. The categories of AI listed in Table 2 are some of the most common methods that use

estimated load profiles to generate optimal heating schedules for individual and aggregated hot water cylinders.

2.2.4 Linear Programming (LP)

This mathematical programming technique is popular for solving complex planning and scheduling problems, and has been well-used in several areas of the power industry [51]. The method finds the optimal value of an objective function in a system in terms of the linear relationships between objectives, constraints and objects. Once a problem is formulated mathematically, Linear Programming (LP) employs one of several popular commercial algorithms (such as CPLEX[2] and Gurobi[3]) to search the problem space for solutions [28]. The ability of LP and related techniques, including Mixed Integer Linear Programming (MILP) to handle complex problems of many thousands of variables at once is a significant strength of this approach. The requirement to employ an external, online or offline solver suits the technique to centralised control authorities with large computational resources, but not to implement on small, stand-alone microcontrollers [52]. For this reason, it was not considered a suitable approach to use in this body of research.

2.2.5 Model Predictive Control (MPC)

Model Predictive Control (MPC) is a popular optimisation approach described in the literature owing to its widespread implementation in controlling many large-scale industrial processes. The approach is a control technique that makes predictions of the system's future state based on an internal mathematical model of the system as it evolves along a moving horizon. Leeuwen et al. [32] describe MPC as more of a "methodology than a single technique". This method of control can work in real time and quickly respond to unexpected changes that impact the system. On the other hand, the approach is also challenging to scale as it requires the development of a detailed mathematical model for every system, including system identification and state estimation stages. Given its large computational requirements and the related

[2] IBM ILOG CPLEX Optimization Studio is an optimization software package.

[3] **Gurobi Optimizer** is a prescriptive analytics platform and a decision-making technology developed by Gurobi Optimization, LLC. The Gurobi Optimizer (often referred to as simply, "Gurobi") is a solver, since it uses mathematical optimization to calculate the answer to a problem.

time cost associated with the complexity of implementation across multiple systems, this approach was also ruled out for further examination.

2.2.6 Artificial intelligence (AI)

AI optimisation methods cover a wide range of separate and distinct algorithms grouped here, such as genetic algorithm, particle swarm and ant colony algorithms. Many of these algorithms draw some inspiration from phenomena seen in the natural world and feature stochastic optimisation across populations to span a search space [53] for optimal solutions. The actual working of these algorithms is advanced, and the black box nature of some algorithms made them too exotic to be considered for consideration in this research.

2.2.7 Reinforcement Learning

Reinforcement Learning (RL), alongside supervised and unsupervised learning, is one of the three main branches of machine learning. RL can be implemented as a model free method, meaning the system under investigation does not need to be mathematically modelled as in the case of MPC. Instead, over many iterations, the behaviour of the system is learned. Rewards and penalties assigned to particular behaviours allow the algorithm to discern which specific actions deliver the best performance characteristics and in a given scenario should be preferred, allowing the conditions leading to positive results to be favoured and made more likely.

2.3 Scheduling Approaches

The approach used by the Bonneville Power Administration Project [54] is a simplified example of scheduled approaches to the hot water controller problem, which focuses on how best to heat a hot water cylinder using the least amount of electricity.

Of the many ways this can be achieved, the scheduled control approach appears to be the simplest because it deals with individual hot water cylinders and requires almost no modification of the hot water cylinder.

The most straightforward scheduling approaches rely on estimating the most appropriate heating times during the day based on perceived patterns of behaviour observed in the household under study. For example, Booysen and Cloete [55]

describe the manual assignment of daily heating schedules based on observed water usage patterns and stated comfort criteria of household occupants. Even with the simplicity of the approach, the authors claim that energy savings of 29% were observed and similarly, Roux et al. observed savings of 9-18% using a scheduled control approach on South African households [56].

More sophisticated studies, however, turn the scheduling problem into optimisation formulations in which an objective function minimising electricity cost is evaluated. Scheduling problems like these are usually solved using non-linear optimization methods [57]. Model predictive control methods and dynamic programming [30] comprise the bulk of the studies for controlling individual electric hot water cylinders observed in the literature.

Unfortunately, both approaches can be computationally intensive and easily become intractable over long time spans or large state spaces. This is because the large number of states to be explored results in an overwhelming number of permutations and leads to the "curse of dimensionality" [58].

One class of solution that attempts to minimise the intractability problem associated with the 'curse of dimensionality' related to dynamic programming methods uses a graphical search approach to calculate the shortest path through the search space. Directed Acyclic Graph (DAG) solutions can be relatively straightforward and computationally less expensive since they do not explore the entire state space [25]. The shortest path graph search using Dijkstra's algorithm [52] is an example of this approach. Here, an investigation of daily hot water demand is used to determine the optimal response to a dynamic price signal to reduce cost and energy consumption. The work in [59] develops this idea further by using an A* search algorithm to conduct a forward search of the state space to find the most efficient scheduling of hot water heating to minimise costs to the user [31].

2.4 Survey of New Zealand Literature

The subject of advanced hot water control has featured in New Zealand academic literature but has become much more prominent in the last decade as awareness of New Zealand's energy supply and the effects of climate change have grown.

The New Zealand Electricity Department, in collaboration with the Department of Statistics, conducted the first thorough attempt at understanding household energy use in New Zealand households between 1971 and 1972 [60]. Focussing on electricity use, the study found that New Zealand households of the time expended on average 44% of their electricity consumption on domestic hot water heating [60].

A more thorough and far-reaching study of the energy consumption and living environment was carried out between 1997 and 2005 by the Building Research Association of New Zealand (BRANZ) [61]. Lasting ten years, this work examined the use of all forms of energy in New Zealand households from multiple sources, including detailed measurements of four hundred households spread throughout the country [61]. With the stated goal of understanding "how, where, when and why energy is used in New Zealand homes", the Household Energy End Use Project (HEEP) provided the first large-scale, robust analysis of energy use in modern houses in New Zealand.

As early as 2000, Wezenberg [62] was also already discussing the design of artificial neural networks to predict hot water demand and how this technology might be incorporated into residential hot water control systems.

Starting around 2010, Gyamfi [63] sought to understand the management of peak demand by investigating DR in a single community in Christchurch. Around the same time, Lapthorn and Watson [64] assessed the potential for smart grids in New Zealand, as did studies by Nair and Zhang [65] and Strbac et al. [11].

In the following years, various studies, such as Croft [66] and Lee [67], have examined the effects of energy storage devices including hot water in microgrids and future implementations of the New Zealand grid featuring more renewable energy generation.

This growing awareness led to, from 2014 to 2018, the GREEN Grid Project [68]. This was a large scale, collaborative project which highlighted the possible benefits that modern renewable technologies and demand response strategies might bring to the

New Zealand electricity system. As the most significant attempt to document the household energy consumption habits of New Zealanders since the HEEP study conducted by BRANZ, emerging from this work was a series of papers that have made preliminary steps in assessing the potential of hot water cylinders to be useful in demand-side management, such as [21] and [69], [70].

Especially important for this research has been the work of Jack et al. [70], which details the modelling process for simple and stratified hot water heaters.
Khan et al. also identified domestic hot water as a "significant candidate" for demand-side management in New Zealand, based on data from 22 houses in the New Plymouth tranche of the GREEN Grid survey [21]. Notably, this study also established the dominance of water heating on seasonal electricity use profiles in New Zealand, showing the potential benefits of moving from a ripple control to more intelligent demand-side management of hot water using modern hot water cylinders and smart control.

Dortan, Carsten et al. [69] have used GREEN Grid load profile data combined with BRANZ's HEEP data to estimate peak period loads from three kinds of thermostatic loads: electric water heaters, heat pumps and refrigerators. The study examined the contributions of each to the national load profile and the peak demand savings under three scenarios, including full curtailment, half curtailment and demand response.

Also utilising GREEN Grid data, Jack et al. [70] discussed the development of one- and two-node hot water cylinder models for accurately modelling hot water cylinders operating under DR in New Zealand. The work in this paper focussed on predicting the demand of individual hot water cylinders during business days as these were the days most likely to see network peaks [70]. The same study was limited to data from seven hot water cylinders. It noted the possibility of expanding future studies to model more representative networks locally and nationally. Possible applications of this approach include the ability of households to respond to pricing signals, participate in demand response programs, and evaluate potential changes in household self-consumption by shifting hot-water demand [70].

Further studies to use hot water cylinder data emerging from the GREEN Grid data include assessing machine learning approaches to predicting household hot water load

profiles [71], a likely step towards managing residential hot water loads with automated smart scheduling.

In a slightly different vein, in [72], Rehman uses hot water data from the GREEN Grid study to analyse the accuracy of machine learning algorithms for Non-obtrusive Load Monitoring (NILM) of residential load profiles. This work attempts to prove the feasibility of utilities learning to understand better or predict future residential load profiles using data recorded remotely from a residential customer's smart meter. No attempt is made to quantify the effects of hot water load control in the broader network.

Other studies have focused on the effects of other technologies, including PV, EVs, heat pumps and BESS [73], [74]. Similarly, several studies have focussed on ways of implementing DR in New Zealand networks with strategies including hot-water control subsumed within the greater context of either household load control or load control at the distribution transformer [16], [75], [76].

Tulabing et al. have also detailed a mock-up of a residential system in two unoccupied houses with simulated household loads [77]. The paper demonstrates the effectiveness of signalling household loads to turn on and off at periods of peak demand to limit demand on the local transformer. Hot water loads are included in the study among other typical household loads, but no optimisation of hot water load control was attempted.

Separate from academia, recent years have seen the emergence of several industry-led working groups and organisations. By collaborating with other key industry players such as the Electricity Authority (EA), these groups are sharply focussed on preparing for accelerated electrification and are championing the benefits of integrating DER into the New Zealand electricity grid [78].

Well-connected groups such as the Flexibility Forum and FlexTalk, have successfully raised the profile of DER technologies and are actively engaging with various agencies including the Ministry of Business, Innovation and Employment (MBIE), Transpower, Ara Ake and multiple EDBs.

Emphasising a “learning by doing” [78], [79] approach, these groups have done much to popularise the potential of DER technologies as useful sources of DR flexibility in the New Zealand electricity grid of the near future.

2.5 Commercial Development

Several products of varying levels of sophistication have been developed over the years, but few have enjoyed widespread success in New Zealand or overseas. Most early commercially available hot water controllers have been small self-contained units easily retrofitted to any hot water cylinder. The focus of such devices has been to reduce the energy use in the cylinder and reduce costs to the consumer.
A commercial literature review reveals that many newer devices, such as [80] are equipped with network communications capabilities through Wi-Fi or cellular technologies and can respond to price signals from local electricity suppliers. A further innovation is occurring with standardised communication protocols such as OpenADR[4] and CTA-2045[5], where hot water controller devices can respond to offers and requests from the system operator to actively participate in demand-side management programmes in real time operation or emergencies. Both protocols are now also under investigation for their potential widespread use in New Zealand [79].

These developments are apparent in recent pilot studies in overseas countries such as the United States and Australia [81], [82]. For example, the pilot study conducted in [54] focussed on the use of ‘static’ strategies [26], including simple on/off control and generic heating schedules. More recent studies such as [82] have gone the next step by adopting internet connected hot water cylinders such as Rheem Manufacturing Company’s[6] active hot water control system [83] to aggregate 2400 South Australian households. The study hopes to show cost and efficiency savings to its participants and use the combined effects of aggregated control to provide useful real-time benefits to local distribution networks and the broader electrical grid [82].

---

[4] Open Automated Demand Response (OpenADR) is an open and interoperable information exchange model and emerging Smart Grid standard.
[5] Modular Communications Interface for Energy Management (ANSI/CTA-2045- – Consumer Technology Association®)
[6] Rheem Manufacturing Company is a privately held manufacturer from the United States that produces residential and commercial water heaters and boilers, as well as heating, ventilating and air conditioning (HVAC) equipment.

2.6 Summary of Literature Survey

A systematic literature survey was conducted across academic, commercial and electricity industry sources. Most recent academic research in this area is concerned with AI solutions to the problem of optimal large scale individual hot-water cylinder control, whereas recent pilot projects conducted by electricity producers have relied on simpler static heating approaches or traditional linear programming methods using commercial optimisation solvers. The most current commercially available devices are small and can be retrofitted to hot water cylinders. The most advanced of these are no longer only trying to reduce individual energy use. They can also communicate, receive information on pricing and signals surrounding the grid state, and adjust their condition to accommodate these signals.

2.7 Research Gaps

So far, there have not been any direct studies on the use of hot water in New Zealand at the local low voltage (LV) network level. This is likely because, until the publication of GREEN Grid survey data, there was no publicly available source of measured household data at the individual household or circuit level. Where predictions of the effects of hot water use are made, they are often done using simulated datasets, which struggle to model the true variability of human behaviour adequately.

Similarly, since hot water control through ripple control is already well established, the perception remains that it is of little value as a resource worth exploiting for effective demand side management [84].

Much of the current focus has instead been on the growing popularity of EVs and the degree to which they will influence peak demand periods on the grid [84], [85], [86], [87]. In contrast, very little work has been done to understand how a modernised use of hot water control might ease peak demand. Similarly, no study quantifies what an alternative form of direct load control to ripple control might look like or what effect it could have on LV networks.

Finally, much has been said about the potential of improved hot water control to contribute to the stability of the electrical grid in the future. However, apart from the

ripple control system, there are still very few studies that actually demonstrate the 'technologies required to implemented demand response at the residential scale' [69].

This study cannot claim to do so either but is a first step on the part of one person to do so using the New Zealand specific data available in the public domain.

2.8 Research Questions

A general trend in the literature surrounding developments in the New Zealand electricity system is recognising the need for new approaches to meet the challenges posed by rapid electrification of a grid increasingly reliant on intermittent renewable energy sources.

Perhaps because of a lack of available data, a more detailed quantification of the 'specific impacts' [88] these changes will bring is still an open question.
An increasing number of studies will likely attempt to fill these gaps. Still, as new technologies emerge, there is a risk that the promise of exciting new technologies will distract from less glamourous solutions that are still to be fully exploited.

With this thought in mind, this study aims to refocus attention on the humble hot water cylinder. This storage device is already widely available in New Zealand, is easily manufactured and disposed of, and can make important contributions to this country's electrified future.

Based on the assumption that the true potential of residential hot water storage devices has yet to be fully realised, even after half a century of ripple control, the following questions will be investigated:

1. *What is the simplest hot water cylinder model that can accurately represent the real performance of real-world electricity hot water cylinders?*
2. *To what degree do more complex models of hot water heating benefit the design of hot water cylinder control?*
3. *What would be the advantages and disadvantages of a simple algorithmic control?*

4. *Are shortest path planning algorithms such as Dijkstra's and the A* search algorithms suited to hot water control at the individual hot water cylinder level?*
5. *To what extent would optimised hot water control benefit the respective stakeholders in an embedded network: the distribution network operator, embedded network owner and end-use consumer?*

Chapter 3 GREEN Grid Data

Knowing that New Zealand must drastically increase the scale of its electricity generation by using intermittent renewable generation[7] technologies such as wind and solar, the debate around what 'systems and services' [89] will best suit the country's unique circumstances has already begun and is ongoing.

More generation capacity will be required to meet forecast demand, but must be generated from renewable sources [89]. The scale of the challenge, as much as a 68% increase in annual electricity consumption by 2050 [89], means the country will be unlikely to meet this level of demand using its traditional approach of ramping up generation [84].

New pressures placed on the electricity system by an increased reliance on intermittent renewable sources of generation will require new approaches and technologies to manage demands on the network [89].

A more detailed understanding of load's nature and characteristics will be required, challenging how loads are analysed and understood in more flexible networks with higher levels of distributed renewable generation. More visibility of residential load behaviours on the LV network will enable a more accurate assessment of how ordinary consumers will likely affect the system's overall stability.

In this chapter, a description will be given of the first detailed attempt made in New Zealand to capture realistic, long-term household data, the strengths, and weaknesses of this dataset, and how it might be used to assess the likely behaviour of LV networks on the grid in the future.

3.1 The Status of Residential Loads in the New Zealand Grid

In line with other developed countries, the residential sector in New Zealand accounts for approximately one-third of the country's total electricity consumption [90]. Unlike electricity consumption in the commercial and industrial sectors, which tend to be

[7] Intermittent renewable generation refers to natural sources of energy such as wind and solar that may not be available on demand. As the climate change commission have observed, "when there's no wind or sun, they have no output" [4].

predictable in size and profile, the nature of residential load profiles, in general needs to be better understood at the individual household level. In particular, there is a large amount of volatility owing to the differences between normal and peak demand [21]. The volatility of residential load profiles also strongly influences network design, as individual behaviours combine to exert pressure on residential networks [21]. These must be engineered to cope with the most extreme demand events [89]. In response, there is a growing trend overseas and in New Zealand to re-assess the nature of residential loads and investigate whether their volatility can be harnessed to contribute to demand-side management strategies that will become increasingly important [84].

3.1.1 Historical Approach to Meeting Demand Pressures

In an electricity system with abundant generation capacity, variations in peak demand throughout the day are catered for by increasing and decreasing supply to match demand. Except from the constraints imposed by dry years [8], this has always been a straightforward process in New Zealand due to its flexible base load supply of hydro-electric power [84].

In a future renewables-heavy grid, scaling up supply to meet immediate demand may not always be as feasible as with carbon-intensive technologies such as gas-peaking plants. Instead, at crucial times of the day, when intermittent supply cannot match demand, demand-side management technologies will be needed to shift demand to times of the day when renewable energy is available.

The ability to change patterns of consumption according to the state of the wider electricity system is called ‘flexibility’.

Flexibility is defined as the ‘the modification of generation injection and/or consumption patterns, on an individual or aggregated level, often in reaction to an external signal, to provide a service within the energy system [78].

Flexibility will emerge as a vital component of the modern electricity system because of its potential to shift the demand of distributed energy resources (DER) in response not just to the availability of energy generation, but also in response to external signals from the electrical grid [78].

3.1.2 Smart Meters and the Move Towards a Smarter Grid

Unfortunately, DSM technologies cannot yet manage networks at a sufficient level of granularity and with sufficient insight into how individual household loads are best manipulated to achieve optimal low-voltage network performance. Arguably the most wide-spread DSM technology used at scale in New Zealand, ripple control is considered a useful tool in managing peak electricity demand, even if the extent of its reach is not accurately known [15]. Estimated to account for as much as 15% of New Zealand's annual peak demand, the long term fate of this pioneering DSM technology is unknown, as more responsive technologies emerge and offer visibility to EDBs and consumer [15].

Amongst the technologies that will be needed for widespread DSM and DR adoption, smart meters have seen widespread adoption in New Zealand's electricity system. The first steps in this process have seen investments in smart metering technology by energy retailers and EDBs [68]. Unfortunately, the rollout of this technology in New Zealand was haphazard, with no apparent communication standard established [22] nationally and with limited access to the data they generate for DSM purposes [68].

The widespread adoption of capable smart meters offers a solution to the problem of electricity distributors and suppliers not having visibility of the lower-level loads in their networks. However, the early uncoordinated rollout of the technology before general agreement on how it could be most effectively used has tempered its progress. Since 2016, individual customers have been given access to their own energy records [68] . Still, as in other countries, privacy concerns have limited the actual consumption data available in public databases [23], [91].

3.1.3 Residential Smart Meter Consumption Data

There is some irony in the lack of detailed data and research into the behaviour of domestic loads available to help improve the efficient operation of LV networks. The information gap is recognised globally as a hindrance to the development of distributed energy generation and smart grid research [92]. In response, communities have begun to share data and increasingly more open-source databases are now available to encourage research and experimentation with data in the renewables research community [93].

These possess several desirable characteristics:

- Higher resolution data sets (< 1 minute) instead of 30 minute samples common;
- Circuit level measurements inside households that allow the consumption of individual appliances to be tracked and their contributions to the total load to be measured;
- Some datasets also contain real and imaginary power readings;
- Some frequency-level data may also be available in specific datasets [11]; and
- Data continuity – some datasets host recorded data from multiple households for more than two years [12].

Often, the data in these datasets is limited in some way, either in breadth, duration, or quality [92]. That being the case, there are many benefits to using detailed household measurement data with high temporal resolution, over difficult-to-acquire top-down data, or data synthesised from appliance usage statistical and other sources [91].

Many of the limitations are due to the complexity of human behaviour and the difficulty of simulating realistic data that captures these behaviours across large populations. The energy demands of residential households are as varied as the make-up of the occupants themselves, and can exhibit degrees of variability in scale and timing that are practically impossible to synthesise in practice [94]. High resolution data from real-world sources are important because they allow researchers to better understand the aggregated effects of individual behaviours of local networks. The complexities of demography, appliance choice and usage interact in highly complex ways to determine the timing and extent of peak power consumption in local low-voltage power networks.

3.1.4 Residential Household Electricity Datasets

The first publicly available residential dataset, the Reference Energy Disaggregation Dataset (REDD) was published by researchers at the Massachusetts Institute of Technology (MIT) in the United States [94]. Since then, other similar datasets have become available such as those listed in Table 3. Notable Residential datasets. A good list of datasets is also summarised in [93].

Table 3. Notable Residential datasets.

Name	Country	Year	No. Households	Duration	Resolution
REDD [95]	USA	2011	6	19 days	3 sec
DALE [94]	UK	2014	5	655 days	10 sec
SCSC [96]	Australia	2010-14	17000	4 yrs.	10-30 mins
Pecan St. [97]	USA	2009-	>1000	Ongoing	Varied

Datasets such as these are of varying size and quality given the enormous cost and effort required to establish each study. An obvious question arises regarding their relevance to New Zealand conditions, since most are based in the Northern Hemisphere and subject to different standards and more extreme climates than experienced in this country.

For these reasons, arguably the most closely related to New Zealand conditions remains the Australian Smart Grid, Smart City (SGSC) dataset [98]. However, this is also of limited value to New Zealand research since the data is confined to smart-meter readings of only half-hourly temporal resolution [98].

In comparison, the North American Pecan Street Project [97] contains detailed circuit level data at one-minute resolution for more than 1000 households. It also includes PV generation and EV charging data for certain household users. The project is still running, and data is available for academic study on registration. Commercial use of the data is available but requires payment [97]. A good summary of similar databases in other jurisdictions is available in [93].

Until recently, there have been no publicly available sources of residential load data directly transferrable to New Zealand conditions. The GREEN Grid Project, which is the subject of the next section undertook the first attempt to address this issue.

3.2 New Zealand GREEN Grid Project

In 2011, MBIE invested in a six-year program of research to develop a better understanding of how an increased proportion of renewable energy generation might impact New Zealand's LV networks and what tools would be required to evaluate their impacts [68].

What emerged from this initiative is known as the GREEN Grid Project, a multi-disciplinary study that ran between 2012 and 2018. Much of the research in the

project was undertaken collaboratively between the Universities of Canterbury and Otago [68]. Various government agencies and organisations in the electricity industry including EDBs, Transpower and the Electricity Authority (EA) also provided input [68].

The United Kingdom Data Service repository hosts the data collected by the project [99].

3.2.1 Data Collection Process

The GREEN Grid dataset comprises detailed, circuit-level measurement data for 40 households, split evenly between two regional centres in New Plymouth and Hawkes Bay [21].

The project occurred in two phases from 2014 to 2018, and circuit level household data was logged to Grid-Spy monitors connected to individual circuits within each household to allow intermittent power demand to be visible at one-minute intervals [100].

The households recruited into the project were selected from volunteer employees working for local EDBs spread across each location [101]. This fact must be emphasised as it prevents the sample from being seen as representative of the New Zealand population [68].

However, given the differences in climate between New Zealand's northern and southern extremities, that both sets of households are at approximately the same latitude allows them to be treated equivalently [21], [100].

3.2.2 Data Analysis Using MATLAB

The process of understanding the underlying data contained in the GREEN Grid dataset is a difficult one, owing to the large amount of data that was collected over a relatively long time-span of more than three years.

The GREEN Grid dataset comprises some 26.8 GB of data, packaged as 45 individual .csv files, one for each of the 45 houses recorded in the survey.

Each .csv fie contains tabularised data containing a household identifier, combined date and time timestamp, a time zone identifier, a label descriptor of the circuit, and power measurement in watts (W).

Of the 45 houses in the dataset, two houses (rf_01, rf_02) are test cases providing data from researchers linked to the study [102]. Hot water data was available for 37 of the 45 houses in the dataset [102].

Data was captured in two tranches. The earliest houses to have data collected were those located in the Taranaki region from mid-2014 onwards [102]. Data from Hawke's Bay dwellings began to be collected from mid-2015 [102].

The large number of readings collected per house, in some cases over nearly four years duration proved challenging given the degree of data management and data analysis required. Inconsistent circuit nomenclature and practices between household datasets also required attention to facilitate cleaning and analysis [102].

To begin with, data for each one-minute timestep were stacked vertically according to the circuit they were recorded for. This required some data wrangling to create 'tidy' data frames for each house [103], in which power meter readings for the circuits are each stored in a single column, with each one-minute reading allotted to a row [103].

Much of the data analysis and early analysis of the GREEN Grid Project data, such as [101], [102],[104] has been undertaken in the R[8] software package. Unlike that work, this research has instead been conducted using MATLAB, on account of its accessibility and the versatility of the MATLAB/Simulink programming environment.

The steps taken to prepare and clean the data included:

- The data comprised of .csv files were imported into MATLAB using the "Import Data" and "readtable" functions,
- Conversion of data to a table and reorganisation to a 'tidy'[9] form. Initially, the raw data consisted of multiple rows for each time step, categorised by the

[8] R is an open source programming language for statistical computing and data visualisation and data-analysis that was developed by Robert Gentleman and Ross Ihaka at the University of Auckland. [105]
[9] Using the 'tidy' data paradigm for tabulated data as championed by Wickham, "each variable is a column, … each observation is a row, and each value is a cell"[106].

named electrical circuits of the house. To convert these to a tidy format, the data was rearranged so that each time step occupied a single row, with separate columns for the data of each household circuit.

- Cell values were checked for gaps using the MATLAB 'ismissing[10]' function. Where gaps were found for individual cells, the 'fillmissing[11]' function was used to repair missing values by estimation based on prior values.
- Data continuity and the existence of outliers were checked using plots and histogram data.

A significant challenge encountered in this early stage of the data-analysis was the lack of uniformity about household circuit naming and inconsistencies in the time zone applied to the raw data. This was particularly the case for the data recorded in the New Plymouth section of the data.

Once cleaned and organise in this way, the tables were merged with time stamped weather data obtained from the 'CliFlo'[12] database. This allowed for weather data to be acquired from weather stations in New Plymouth and Hastings to be joined to the household data for each household.

Code for the initial data analysis and visualisation of circuit data is presented for reference in Appendix A.

3.2.3 Characteristics of Households

Detailed summaries of the demographics for the collective of houses are summarised in detail in [21] and [108], as reported by occupants in a detailed survey at the beginning of the project [109].

Most notably, all except one of the houses were stand-alone dwellings with sizes varying between 2 to 6 bedrooms [21]. The number of occupants ranged between 1 to 6 occupants. All age groups were represented among occupants, including children, teenagers, adults, and mature adults [90], [108].

[10] https://au.mathworks.com/help/matlab/ref/ismissing.html
[11] https://au.mathworks.com/help/matlab/ref/fillmissing.html?s_tid=doc_ta
[12] Cliflo is an interface provided by the National Institute of Water and Atmospheric Research (NIWA) that allows public access to the National Climate Database for New Zealand. [107]

The electrical circuits for each house generally include a single-phase conductor from the main electrical grid, labelled in most circuits as ‘incomer’. Where appropriate, separate circuits for hot water, heat pumps, kitchen appliances and lighting were also labelled and designated with a unique identifying code. The houses attributed to the New Plymouth group were more complex than the Hawkes Bay residences, including four homes with solar panels.

Of the 43 sets of household measurements, 15 occupants (35%) responded to operating a heat pump for space heating in their homes. In comparison, 29 (64%) occupants reported using electric heating to heat their hot water, slightly less than the national average of 87% reported in the Household Energy Use Project (HEEP) undertaken by the Building Research Association of New Zealand (BRANZ) [21].

Given that the first houses in the New Plymouth cohort began recording data in 2014, no EVs are recorded in the survey data. Similarly, since LED lighting was not prevalent at the time, load data associated with lighting appears higher than what it would for most households today.

3.2.4 Continuity of Data

For this study, hot water load data from each property was analysed where possible in separation from power data from other circuits in the house. This simulated a non-standard water heating profile compared to control free operation.

Of the 40 houses featured in the study, a mix of 21 homes from the New Plymouth and 19 from the Hawkes Bay regions were chosen. These households have the most complete datasets available for the main incomer and hot water circuits from the 40 houses comprising the entire GREEN Grid dataset. This is akin to the approach taken in [100], in which a mixture of houses from the same dataset was selected to estimate the benefits of home battery use on the electrical grid.

Parker [71] also describes a similar process of identification of suitable households with sufficient data from within the dataset, noting also the gaps in data that made certain houses in the dataset unsuitable to study.

Table 4. Household hot water cylinder capacity.

Household	Location	No. Occupants	Cylinder Capacity (L)	Water Heating Method
rf_6	New Plymouth	2	180	electrical
rf_14	New Plymouth	2	114	wetback
rf_22	New Plymouth	2	180	electrical
rf_23	New Plymouth	1	113.5	electrical
rf_24	New Plymouth	4	180	electrical
rf_29	Hawkes Bay	3	180	electrical
rf_30	Hawkes Bay	4	135	electrical
rf_31	Hawkes Bay	5	135	electrical
rf_32	Hawkes Bay	4	135	electrical
rf_33	Hawkes Bay	4	180	electrical
rf_34	Hawkes Bay	3	300	heat pump
rf_35	Hawkes Bay	4	177	electrical
rf_36	Hawkes Bay	3	180	electrical
rf_37	Hawkes Bay	4	180	electrical
rf_38	Hawkes Bay	4	180	electrical
rf_39	Hawkes Bay	4	180	electrical
rf_42	Hawkes Bay	5	180	electrical
rf_44	Hawkes Bay	5	180	electrical
rf_45	Hawkes Bay	5	180	electrical
rf_46	Hawkes Bay	3	180	electrical

Of the selected houses from the list in Table 4, only two (rf_14 and rf_34) do not use resistive heating as the primary means of hot water heating. In Table 4, the respective capacities of each hot water cylinder are also included for comparison between homes.

Unfortunately, even in this subset of the original dataset, not all Grid-Spy monitors were functional for the full duration of the study. Several gaps exist in the dataset that sometimes last for many days or weeks [102].

Visualisations of the continuity of data available for a select group of houses are offered in Fig. 1. Continuity of overall household electricity consumption data between March 2015 and March 2016. Continuous month-long sections of main incomer data between March 2015 and March 2016 are represented by the colour green. For comparison, in Fig. 2, a summary table depicting the monthly continuity for hot water data is shown.

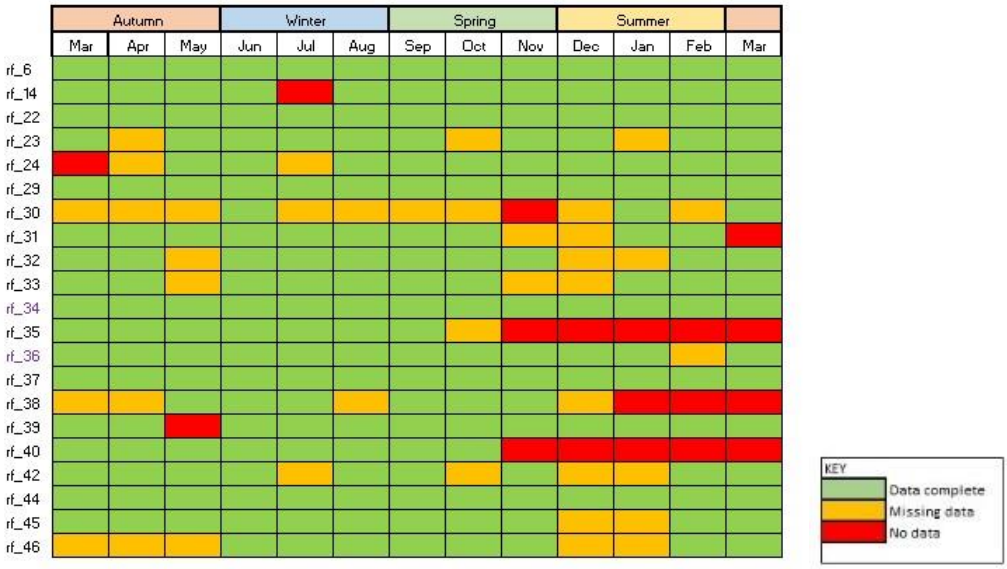


Fig. 1. Continuity of overall household electricity consumption data between March 2015 and March 2016.

Fig. 2. Continuity of hot water data for the twelve months between March 2015 and March 2016.

3.2.5 Discussion of Data

The energy consumption of each household varies widely depending on factors that include the number of occupants, the amount of time they spend in the house, their activities and the kinds of appliances used at the property. As noted by other studies, there is considerable variability between houses [94]. Self-reporting of energy use is also an unreliable indicator of the actual energy use observed [108], [110].

The annual household electricity demand data in kWh is shown in Fig. 3 for the selected group of houses.

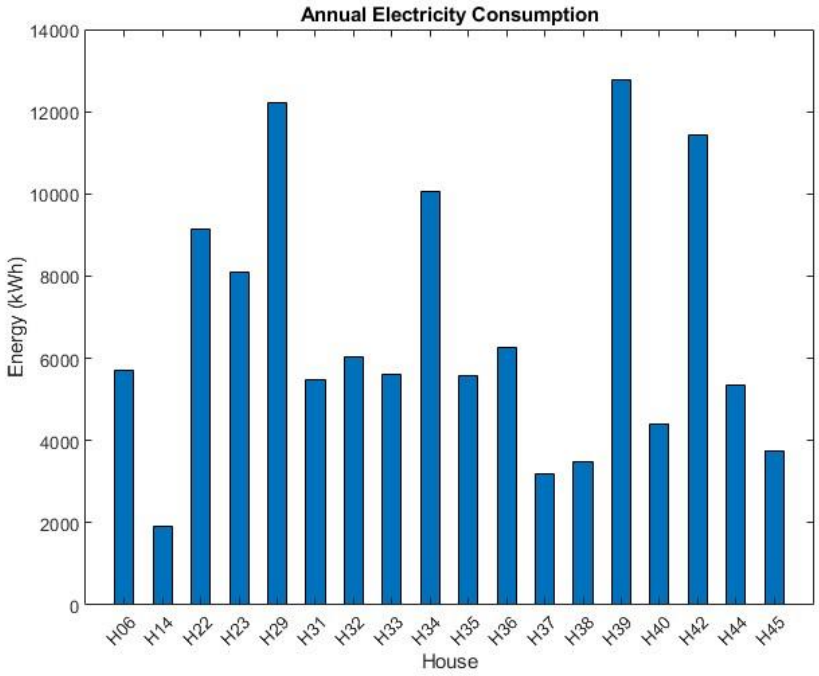


Fig. 3. Comparison of total annual electricity consumption for selected homes.

Of the households represented, there is some correlation between overall demand and variability of energy demand. In Fig. 4, the mean variability in monthly energy consumption is plotted for each house where data is available. Values at the top of each box in the plot indicate the monthly energy consumption during winter, whereas those in the lower half represent the warmer months when less energy is used.

The plot in houses 29, 35 and 39 have the highest mean monthly consumption values. House 29 peaks in July to consume 1600 kWh in that month (55 kWh/day), possibly owing to a reliance on electricity for space heating and water heating in that house.

Also, house 24 was removed to simplify analysis since it contains solar modules and exhibited negative net energy consumption.

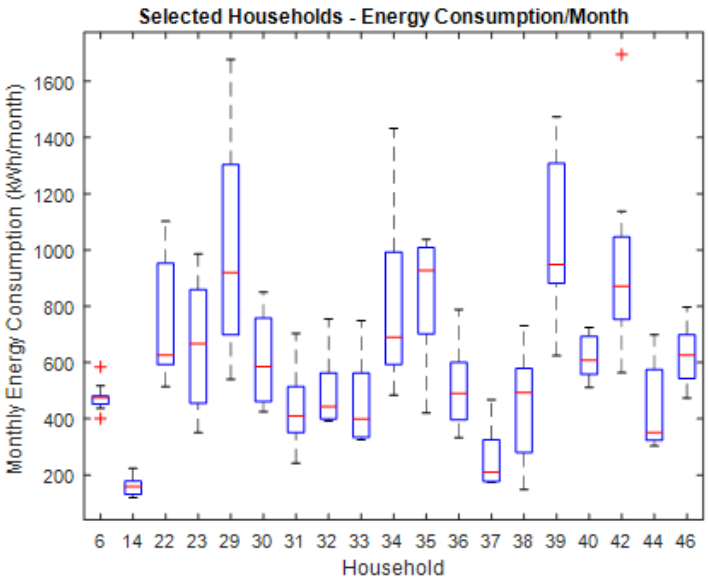


Fig. 4. Box and whisker plots showing variability of monthly household load for select houses in 2015.

This is consistent with the data provided in [61], where the electrical energy used to heat residential hot water per household is estimated to be approximately 2,440 kWh for households across New Zealand.

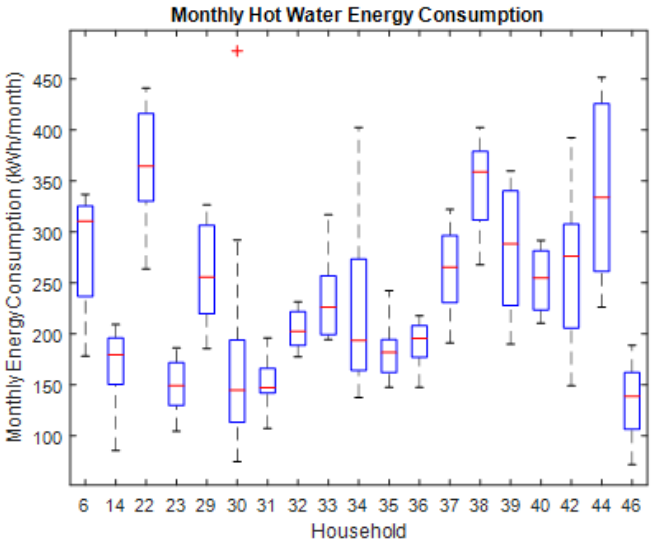


Fig. 5. Box and whisker plots showing the average monthly residential hot water load.

There is less variation between houses in this plot, and the annual variation across months is more consistent. The mean energy consumption per month is within the range of the 30% of household energy consumption attributable to New Zealand

homes [21], but in some months, it can spike to values at or above 50%, such as for houses 14, 22 and 37.

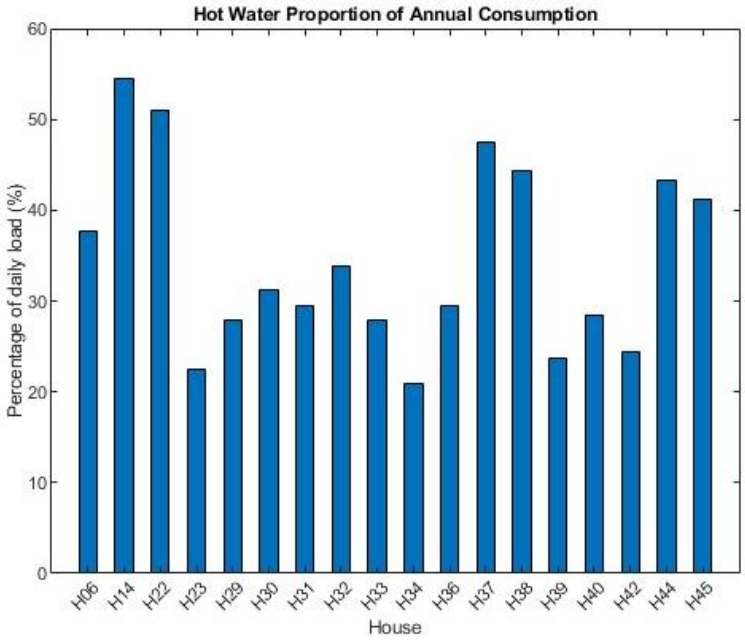


Fig. 6. Percentage of daily household load attributable to hot water heating load (all houses).

Though relatively high, this is also not inconsistent with estimates of the proportion of household electricity for domestic hot water by EECA (39%) and HEEP (28%) [61].

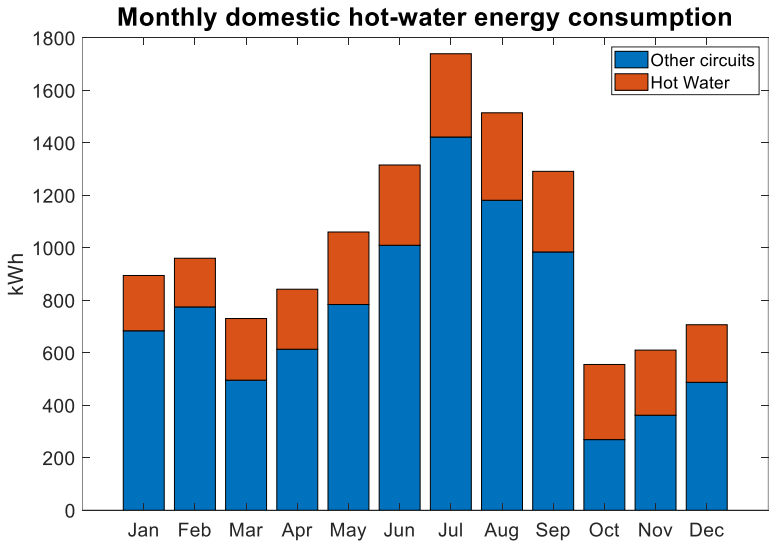


Fig. 7. Proportion of electricity use attributed to hot water consumption - House 29.

The proportion of electricity used for hot water heating by house 29 (rf_29) is typical of the seasonal trends observed across the cohort. In

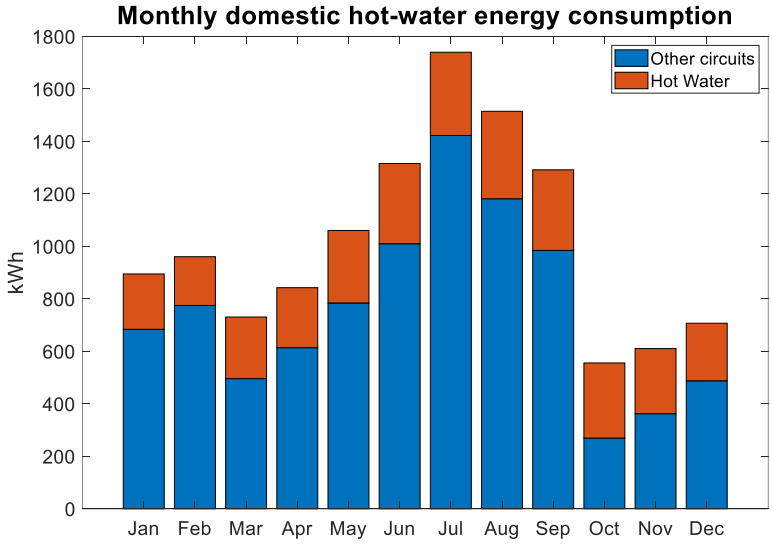


Fig. 7, it can be seen there is a gradual increase in energy consumption during the colder months between April and October and reduced but still significant consumption between November and April.

3.2.6 Individual Household Trends

Given the variability noted between household energy consumption patterns within the dataset, it is difficult to pick any clear predictors of overall energy use and behaviours.

An interesting aspect of the GREEN Grid study was the collection of detailed demographic data by means of long form and short form household surveys [104] [109].

Data collected for the long survey included questions pertaining to the age and nature of the building, the number and make-up of occupants, and detailed questioning over appliances present in the house and characteristics of its use such as frequency and number of times used per day.

A list of the most energy intensive appliances included in the long and short surveys is provided in Table 5. Of the two survey types, the long form survey was not supplied to all participants [109], and as a result, detailed data from the survey is not available for many of the houses, especially those from the Hawkes Bay group.

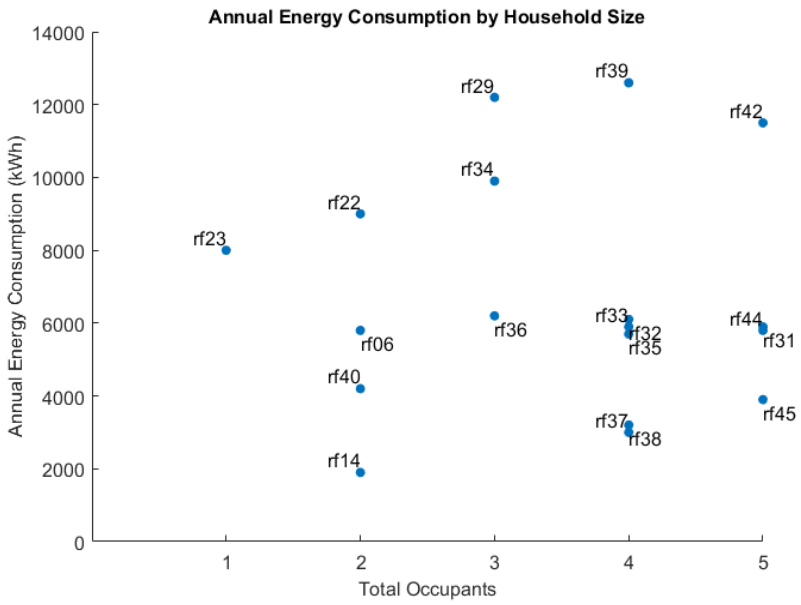


Fig. 8. Annual Energy Consumption by number of household occupants.

With detailed data available for only a subset of households, it is difficult to find any obvious pattern in total energy consumption recorded at each property. In Fig. 7, a grouping of the data by the total number of occupants shows a loose grouping of households with greater than 8 MWh/yr energy consumption, and a second group with energy consumption less than 6 MWh/yr.

What is the most likely predictor of a house occupying the higher energy consumption group is not clear from the information available. For instance, the household with the

highest overall annual consumption is rf_39, occupied by three adults and one teenager. rf_39 is not obviously out of the ordinary if analysed from the perspective of the appliances present in the house. It has three fridge freezers, but no electric space heater. Like many of the houses in the study, detailed characteristics of the property which might indicate the reason for its large energy consumption, such as number of bedrooms, or age of the building are not available because the occupants of this house did not undertake the long survey.

Similarly, information relevant to the two other high consumption households, rf_29 and rf_42 is also limited by the same absence of long form survey data.

Households rf_22, rf_23 and rf_34 on the other hand, also members of the same high consumption grouping, have 1 to 2 occupants respectively, but also feature heated spa pools which probably contribute to their overall energy needs.

Table 5: Survey results of main household appliances for selected households.

House	Survey Form	Oven	Microwave	Fridge/Freezer	Dishwasher	Washing Machine	Dryer	Electric Heater	Heat Pump	Spa Pool
rf_6	Long	Y		1		Y				Y
rf_14	Long	Y	Y	1	Y	Y	Y	Y		Y
rf_22	Long	Y	Y	2	Y	Y	Y	Y		Y
rf_23	Long	Y	Y	1	Y	Y	Y			Y
rf_24	Long	Y	Y	2	Y	Y				Y
rf_29	Short	Y	Y	2	Y	Y		Y	Y	
rf_30	Short	Y	Y	3	Y	Y				Y
rf_31	Short	Y	Y	2	Y	Y			Y	
rf_32	Short	Y	Y	3		Y	Y	Y	Y	
rf_33	Short	Y	Y	2	Y	Y	Y	Y	Y	
rf_34	Long	Y	Y	2	Y	Y	Y		Y	Y
rf_35	Short	Y	Y	2	Y	Y	Y		Y	
rf_36	Short	Y	Y	1	Y	Y		Y	Y	Y
rf_37	Short	Y	Y	1	Y	Y	Y	Y	Y	
rf_38	Short	Y	Y	2	Y	Y			Y	
rf_39	Short	Y	Y	3	Y	Y	Y		Y	
rf_42	Short	Y	Y	2	Y	Y	Y		Y	
rf_44	Short	Y	Y	3	Y	Y	Y		Y	
rf_45	Short	Y	Y	2	Y	Y	Y		Y	

3.2.7 Case study – House 29

An annual plot of the average power demand of House 29 is shown in Fig. 9 to give a sense of the general shape of demand throughout the year.

House 29 contains three occupants, two adults and one child. In the survey questions accompanying the GREEN Grid survey, the occupants report moderately high energy bills. Their response to the GREEN Grid survey questionnaire shows they operate two fridge-freezers, a dishwasher, oven and microwave. This household also uses heated towel rails, but most notably uses electricity for space heating and water heating, the two key indicators of high energy use [70].

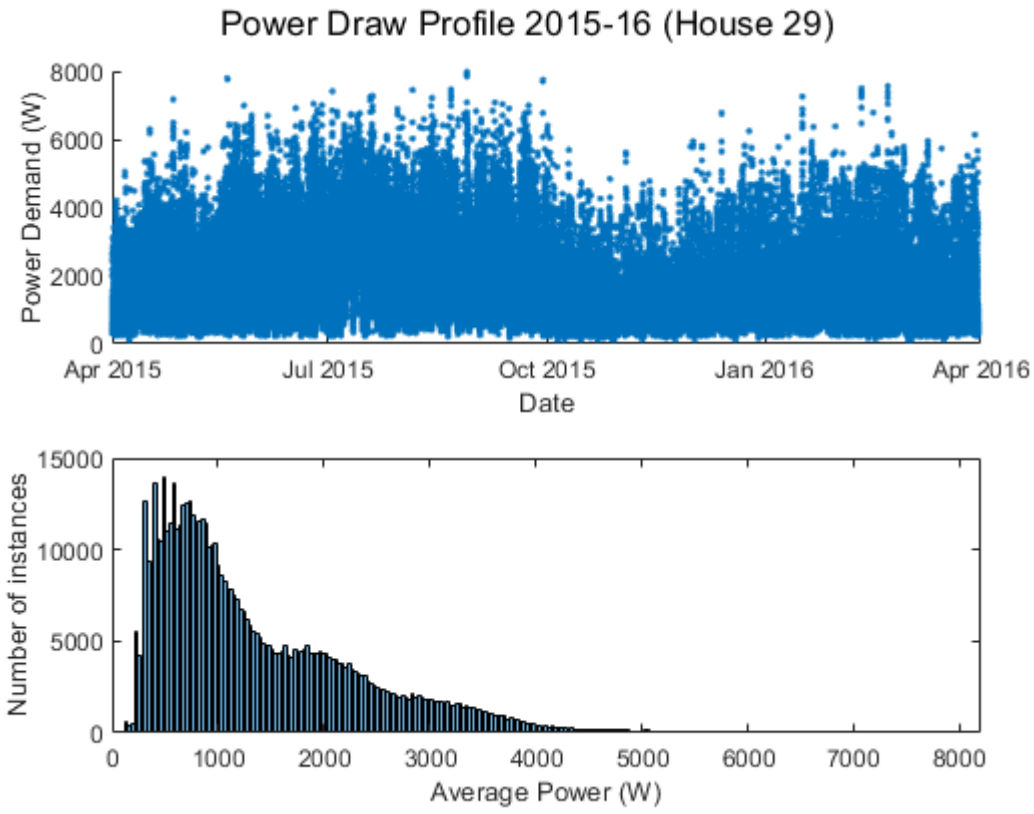


Fig. 9. Annual household load - House 29.

The lower plot in Fig. 9 is the histogram of average power levels demanded from the grid over the same timeframe. It shows that most of the power drawn is below 2kW, and higher peaks account for only a small percentage of the power used.

Analysis of the annual power consumption patterns of House 29, using a Multiple Seasonality Decomposition using LOESS (MSTL)[13] analysis described in [111] and [112], is carried out to show a clear seasonal effect on energy consumption. This approach reduces the chaotic looking daily load profile to its underlying components. For house 29, the trend for energy consumption peaks in mid-winter (with the coolest weather) and decreases steadily until stabilised by the warmer weather of early summer.

A long slow increase in power consumption is evident through the spring until the arrival of winter when the cycle repeats.

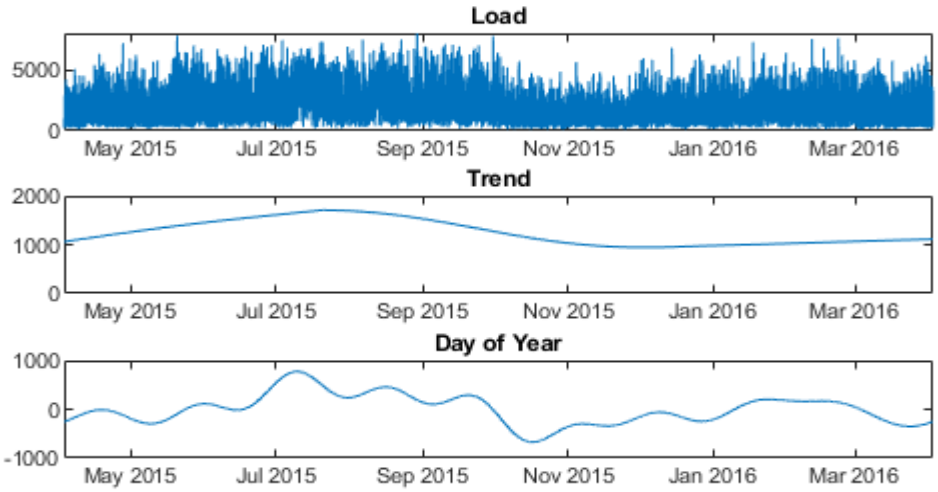


Fig. 10. Seasonal decomposition of annual household load to identify 'trend' and 'day of year' energy consumption patterns.

Likewise, the overall pattern of daily energy consumption varies with the 'day of year'. The days with the highest household energy consumption are correlated with the coolest weather experienced between July and October. However, the ‘day of year’ periodicity also shows some increase in daily energy consumption during the hottest months from January to March, which with this household, might be attributed to the presence of a heat-pump, which presumably consumes energy for cooling.

As visualised in Fig. 11, less than 10% of the power drawn throughout the year occurs at levels above 3 kW. This trend is seen in all households and aligns with the other studies such as [70], in which the daily residential peak is described as ‘the most

[13] MSTL is a statistical algorithm for time series analysis that extends the traditional Seasonal-Trend decomposition method (STL) for the decomposition of time series with multiple seasonal patterns [111].

volatile', compared to profiles typically seen in industrial and commercial sectors throughout the year.

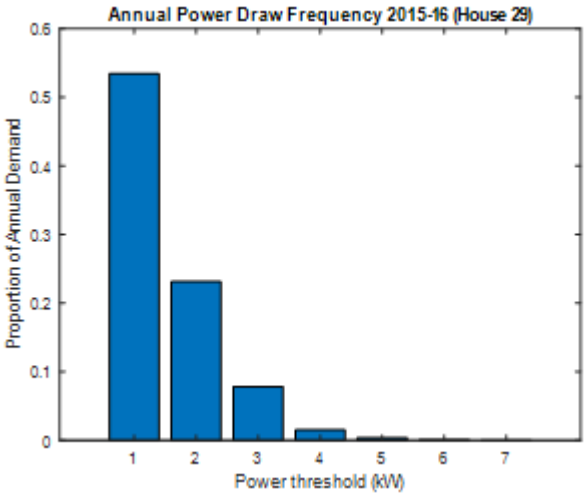


Fig. 11. Histogram of power draw values - House 29.

It also illustrates the challenge of designing electrical infrastructure for residential customers, with extremes between normal and peak demand on any given day. The large variance between 'normal' and 'heavy' power draw for individual households is likely to be exacerbated with the mass adoption of EVs in the near future [85].

3.2.8 Diurnal Pattern of Household Peaks

The contribution of hot water load to the daily peaks is plotted in Fig. 12 to show how closely aligned typical hot water demand is with the daily routines of the occupants of the house. Three consecutive weekdays in June are shown, with clear diurnal peaks in the morning and evening. These are consistent with the work of Khan et al, who identified New Zealand residential load profiles as typically having one peak in the morning (T2) and another broader peak in the early evening (T6) [5].

The same daily diurnal pattern is seen in Fig. 13, which more clearly shows plots for the same three days for House 37, since this household does not carry such a high base-load demand as seen in House 29.

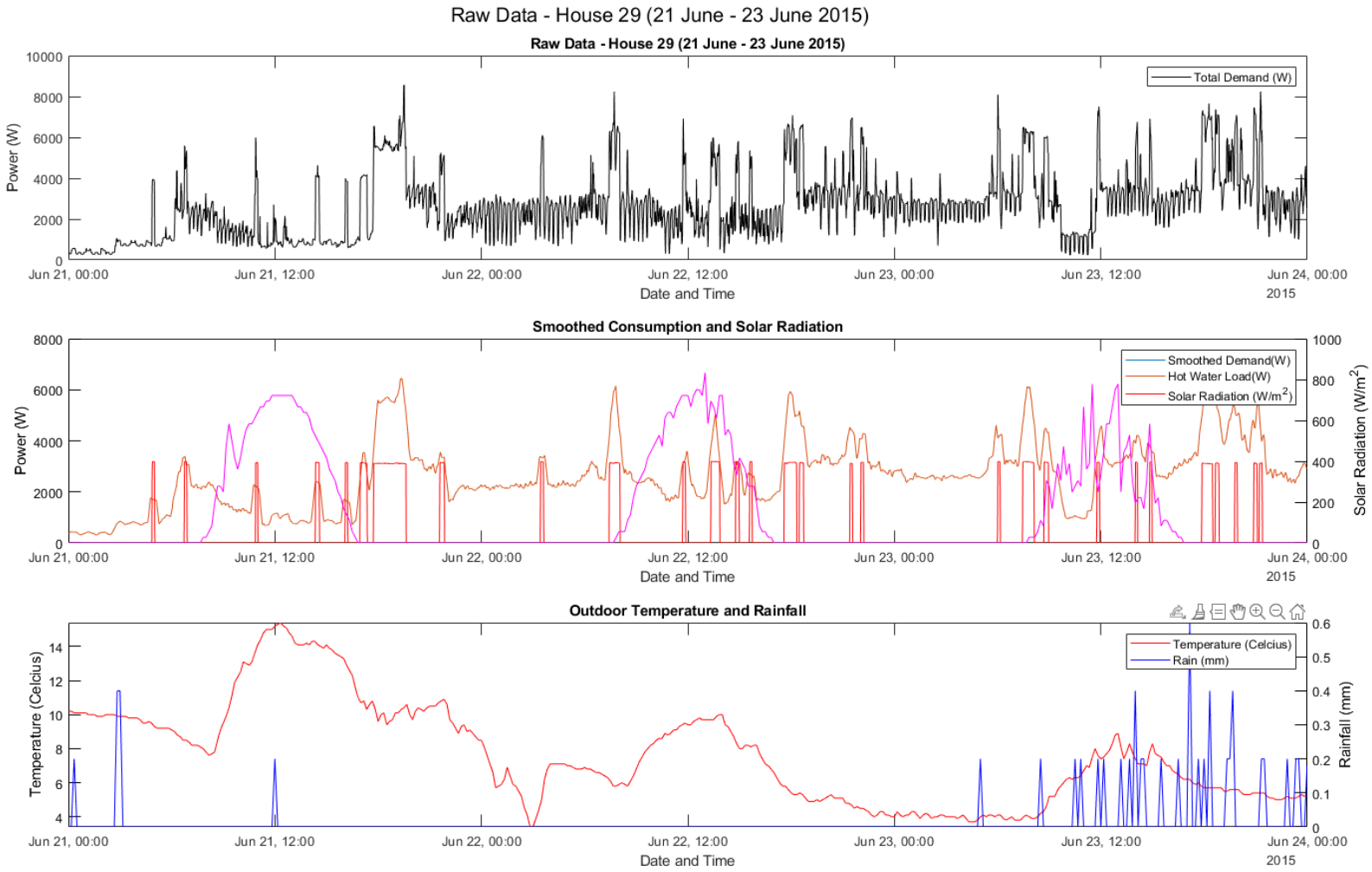
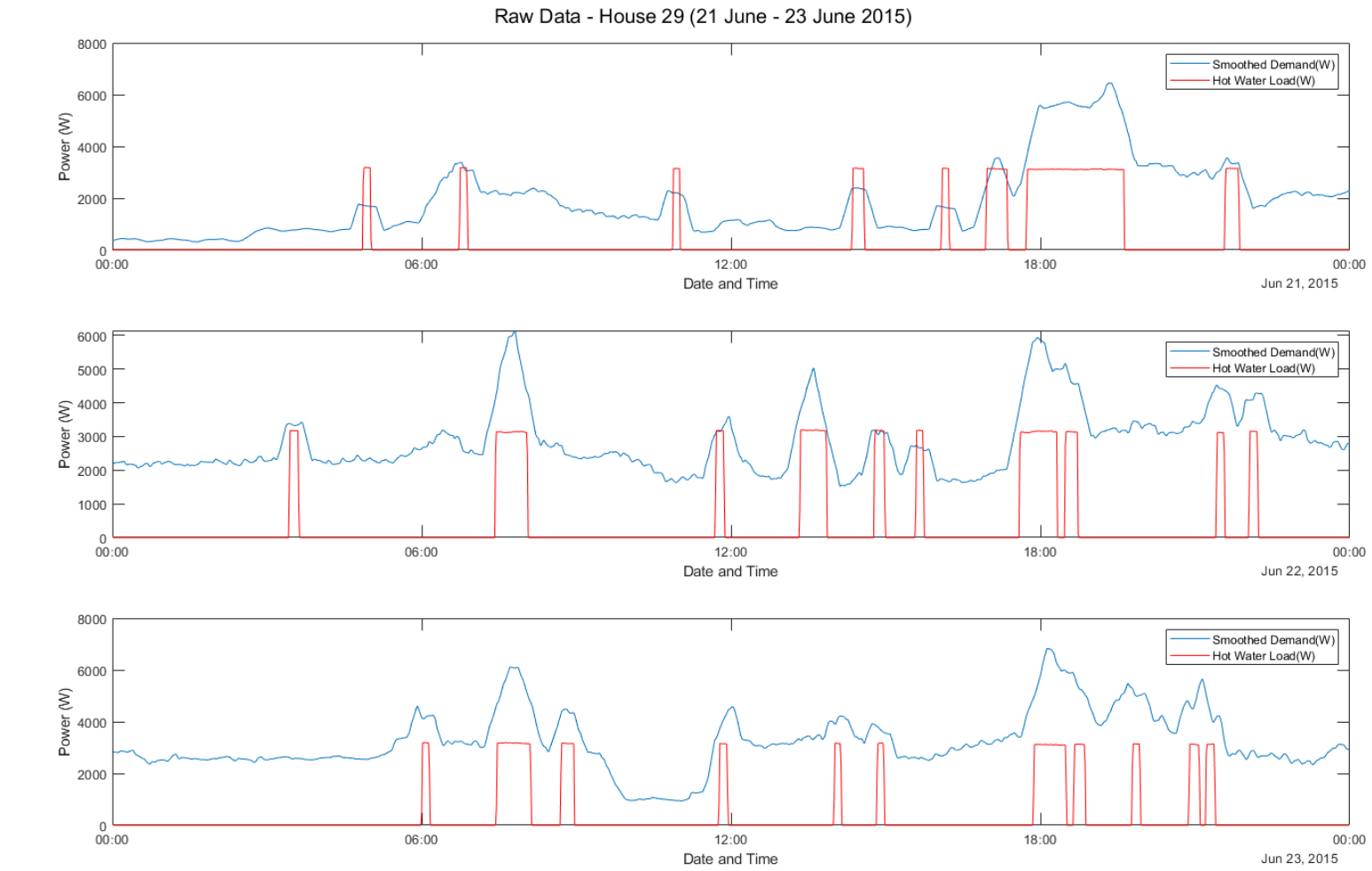


Fig. 12. Hot water contribution to incoming power draw for three consecutive days - House 29.



.

Fig. 13. Hot water contribution to incoming power draw (red) for three consecutive days - House 29.

Of interest in both cases, is the degree to which the morning and evening peaks of the houses are influenced by peaks associated with occupant morning behaviour, as was also noted by Jack et al. in [6].

An analysis of household data using, again visualised by MSTL decomposition shows in Fig. 14 the clearly diurnal pattern exhibited by all households in the study, strongly reflects the daily household energy use patterns of many households in New Zealand.

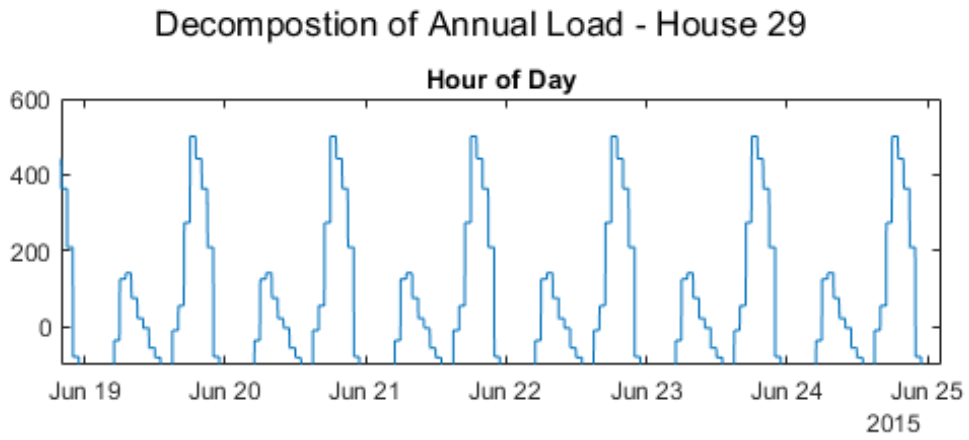


Fig. 14. Diurnal periodicity of household energy consumption for a week in June - House 29.

This trend is also consistent across similar plots for all 21 households and is reinforced by an analysis of average hot water usage periods throughout the year.

In Fig. 15, a heat map of the average power consumption due to hot water usage is shown for an entire year. The data, in this case, has been normalised to account for the differing heating element power ratings between household hot water cylinder models. What is apparent from this treatment of the data are the two dark blue horizontal bands present. Even averaged over twelve months of use (weekends included), there is clearly a morning and evening period of hot water usage common across all households.

The potential to shift these periods of use to other times of the day, especially the early morning and to a lesser extent, early afternoon that offers the promise of hot water control and the path to a more flexible grid.

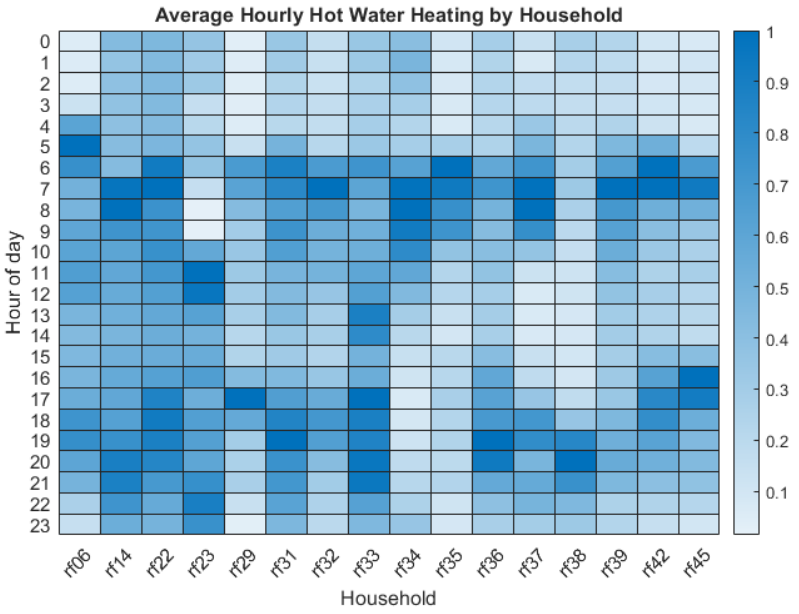


Fig. 15. Average normalised power draw for hot water cylinder heating households in one year. Power draw is normalised to account for differing heating element sizes between household hot water cylinders.

3.2.9 Profile Differences between Workdays and Weekends

The simple pattern seen in Fig. 15 is incomplete in the sense that differences seen between working days and weekends are averaged out of the data. The same is true for data caused by seasonal differences, especially between the cold months of winter and more temperate conditions of spring, summer, and autumn.

To this point, Jack et al. [17] have discussed the apparent differences between electricity use profiles on working days and at weekends. Their observation is that working days are more likely to experience peak loading events severe enough to stress distribution networks, whereas loads are spread more evenly during weekends and avoid this risk. The plot in Fig. 16 corroborates this concern, where the daily power profiles averaged over the winter months between June and August are shown for multiple houses selected from the dataset.

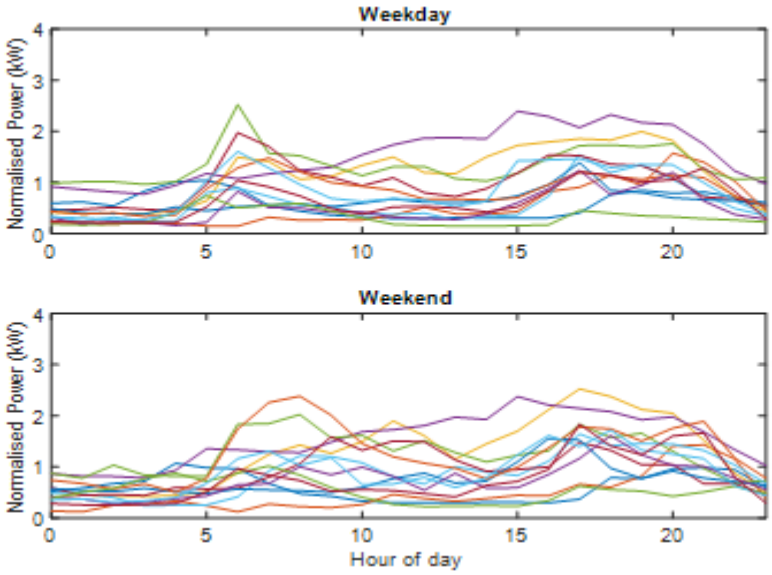


Fig. 16. Comparison of average weekday and average weekend electricity consumption profiles for GREEN Grid households.

Only one house in the grouping exhibits an equivalent level of peak load on weekends compared to working days during the expected period of highest demand. The peak loads of the other houses in the group, particularly in the morning, are delayed in comparison, and their peaks spread through the later morning past 10:00 am, reducing the average morning load during the T2 period between 5:00 and 9:00 am.

The same effect is seen in Fig. 17, showing the seasonal decomposition of household electricity consumption using MSTL [111] for the annual load of House 29 as an example. The stepped nature of the plot, shows peaks occurring periodically on a weekly basis, coinciding with Saturdays and Sundays when this family is presumably more likely to be at home for more hours of the day and thus have higher energy demands.

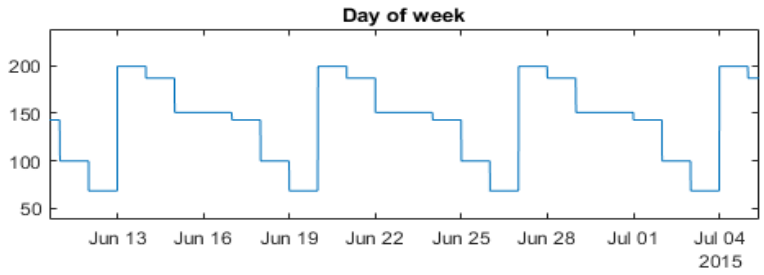


Fig. 17. Seasonal decomposition (House 29) showing periodic consumption behaviour by 'Day of week'.

In this household, weekday energy consumption is lower than weekends, and noticeably lower on Thursdays and Fridays, a pattern that is consistent throughout other months of the year. The nature of energy consumption amongst households, even in a small sample group, follows the same general pattern, yet keeps an element of uniqueness, symbolic of the varied personalities and habits of the householders living there.

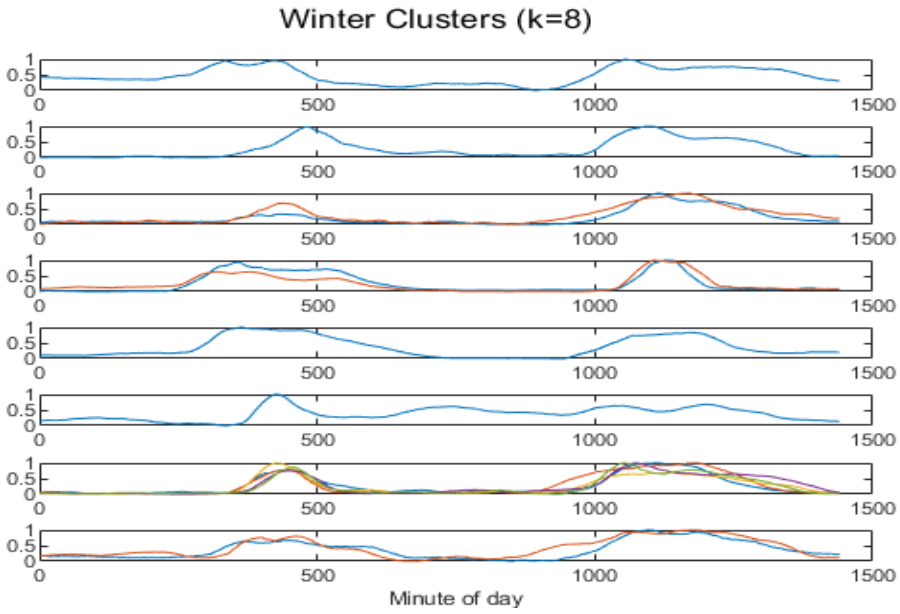


Fig. 18. A weakness of the GREEN Grid data on display here is the lack of diversity between household patterns of electricity consumption. Of the 21 houses selected, in winter there are only eight distinct consumption patterns, which all share early morning and evening peaks.

3.2.10 Limitations of the GREEN Grid Data

The GREEN Grid Project was an important interdisciplinary contribution to the development of knowledge around the patterns of residential energy use in New Zealand [68]. The data acquired in the household energy measurement phase of the project was built on the foundational work of the BRANZ HEEP, by providing for the first-time, high-resolution household data captured across an extended time period.

Even though the data was collected from a small cohort of houses compared to the previous HEEP studies, the GREEN Grid data, sampled as it was at one minute intervals was more granular than the ten-minutes sample frequency of the earlier work conducted in New Zealand to that point [68].

However, even though the wider research surrounding the GREEN Grid project has led to the publication of dozens of papers on the potential of technology to improve the efficiency of the electricity grid in New Zealand, surprisingly few studies have used the detailed one-minute data it provides.

Where one minute load data from the study has been used, its inclusion has been selective, given the lack of continuity and gaps apparent in time series data held for certain houses.

For instance, Mair et al. [100] used a reduced sample of GREEN Grid data from 20 houses to discuss the sizing of domestic batteries, but split these unevenly between houses in the New Plymouth and Hawkes Bay [100].

In his work on NILM, Rehman [72] chose to use thirty days of data from only four households (rf_01, rf_02, rf_36 and rf_42).

Similarly, using GREEN Grid hot water consumption data consolidated into half-hourly average values, Parker [71] used only 22 of the 47 household profiles available in the dataset to compare the effectiveness of predictive models to learn household hot water consumption patterns.

Though anecdotal examples, there are other problems with GREEN Grid dataset that limits its effectiveness as a reliable source of household data to base studies on in New Zealand.

In some sense, the study was ahead of its time, and was carried out in the brief period before the resources of cloud computing were in widespread use. Consequently, all data recorded had to be stored in hardware on site, which entailed specialised GridSpy monitors [104] to be installed on all circuits of the participating houses. This limitation meant that data could not be monitored remotely during the collection process, so when data collection equipment failed, this was not noticed until the time came for the data to be collected from the site.

Similarly, readings from some hot water circuits are very inconsistent, or indicate hot water cylinders not functioning correctly for long periods of time. But these failures

could not be noticed until the end of the study periods when data was downloaded, making the data from some houses of limited value to more detailed study.

Having begun nearly a decade ago, there may also be some scope to question the relevance of some measurements to future household consumption trends. For example, the lighting loads across the dataset are estimated to account for between 7% and 9% of the total household loads [104], which may be high now given the move from incandescent lighting to Compact Fluorescent (CFL) and Light Emitting Diode (LED) technologies in the years since [113].

The manner in which participants were selected for the study also detracts somewhat from the collective value of the data that was collected to fully represent New Zealand households. Unlike the earlier BRANZ study for example, the GREEN Grid project is localised to small in scale. With 47 households involved, the survey size is small and spread across two regions on opposite sides of the North Island, albeit at similar latitudes.

Since all participants were also employees of either of two EDBs, there are questions around how representative they are of the greater population [68]. This is particularly important for understanding a variety of load profile shapes that exist in the wider community, because the behaviour of the occupants of a household is such an important determinant of the quantity and times of energy consumption [114].

Comparison of household profiles generated using a K-Nearest Neighbours[14] approach [115] for example, shows very little diversity in load profile pattern. In Fig. 19, it is seen that most households with two distinct peaks, one in the morning around 7:00 am, and a prolonged peak later in the day between 5:00 pm and 8:00 pm, consistent with full time workers preparing for work in the morning, leaving the house for the day, and returning in the evening.

[14] The K-Nearest Neighbours (KNN)_ algorithm is a popular machine learning technique used for classification and regression tasks.

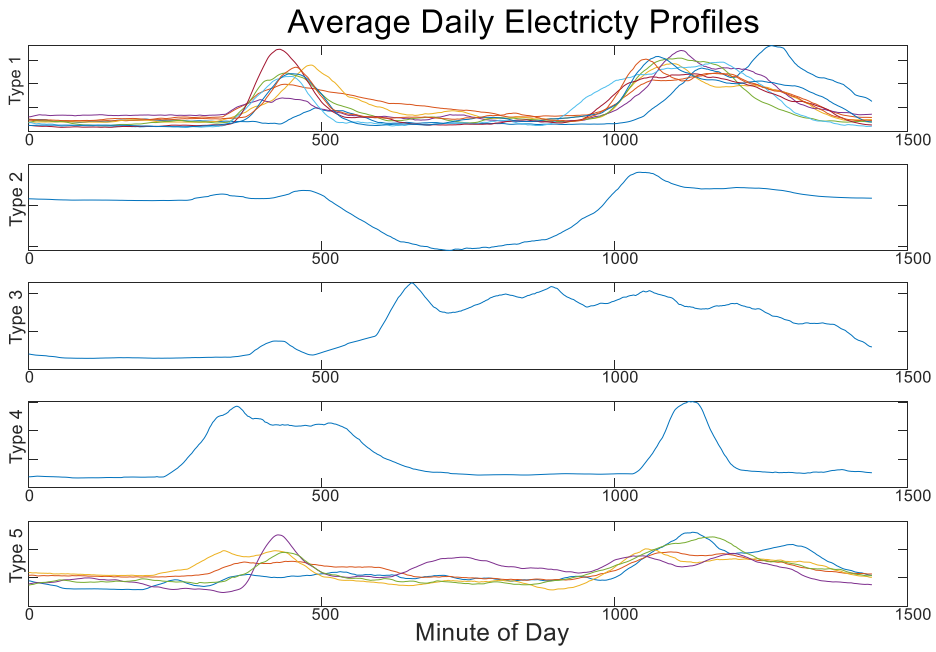


Fig. 19. Categorisation of load profiles from the sample households, showing limited diversity.

Also in Fig. 19, it is possible to see that only one household that on average uses load steadily throughout the day. One profile, the second from the top has a solar module that exports power during the middle of the day. The remaining houses, to some degree have minimal energy use through the middle of the day, and two distinct periods each day of intensive energy use, presumably corresponding with occupancy.

A detailed attempt was made to gather demographic data alongside the technical data provided by measurement equipment. The design of a detailed survey to fill in much of the human element that might explain the patterns of energy consumption is described in detail in [108]. The answers to a selection of the survey questions are available in a spreadsheet available with the downloaded package of data from the UK data repository. Unfortunately, the detailed long-form survey was strictly voluntary for study participants [108], and as a result, there are gaps in the consistency of this important resource as well. This makes understanding the dataset in its entirety difficult, especially for the Hawkes Bay houses, which are the most represented in missing data.

3.3 Summary

The first tranche of data collected in the GREEN Grid data survey is now a decade old. At the time it was proposed, it was an ambitious project that, given the work required to install metering on the individual circuits of more than 45 residential dwellings, represents a massive investment in time and resources.

The data provided by the study has been used in several studies by New Zealand academics to explore the potential challenges of a more flexible electricity grid in the future in this country, but some shortcomings in the data have limited its influence.

These include the limited continuity of the data, questions around how well it represents 'typical' New Zealand households, and the inaccessibility of associated demographic survey data recorded at the time of the survey.

Despite the shortcomings of the dataset, particularly the non-contiguous nature of the data in some households, it remains a valuable resource. There are currently no other sources of data at such fine granularity that capture the daily habits and lifestyles of New Zealand residents that are available in the public domain.

A basic examination of the data conducted for this study has found many of the same patterns noticed in other more detailed studies such as [70], [71], [104], [110].

The overall proportion of energy consumed by heating water in each of the houses chosen, also backs up conclusions reached in studies such as [11], [70], [104], that hot water heating load is well suited to demand response strategies in reducing peak demand in future grid technologies, particularly when carried out as an "automated process" that does not require constant manual intervention [69], as does the ripple control technology employed to this day.

In the next chapter, experimental work will be carried out to characterise the physical properties of a commercially available, residential hot water cylinder. That analysis will assist in the design of mathematical models that simulate the functioning of hot water, and that will help in the process of designing automation into working devices for future development.

Chapter 4 Electric Water Heater Laboratory Test

4.1 Electric Water Heaters as Energy Storage Devices

Outwardly, hot water cylinders are simple devices, whose reliability and plain exterior hides the complexities and subtleties of fluids and thermodynamics. To capture this complexity, many formulations and models of varying degrees of accuracy and computational efficiency have been developed [116]. Chandra and Matsuka mention six main classifications of hot water cylinder model in [117]. Dumont et al. offer no fewer than nine [116], some of which will be discussed in more detail in the next chapter.

Unfortunately, real world validation of these models is challenging, and there are fewer examples in the literature where the predictions of hot water models are extensively tested for real world application [118]. This could be because testing a hot water cylinder in laboratory conditions is a time consuming and expensive process that puts the process of experimentation and validation impractical.

If carried out thoroughly, the slow nature of hot water cylinder heating, with time constants on the scale of hours, requires many weeks of supervision and monitoring.

This section describes the steps the author took to investigate the characteristics of a commercially available residential hot water cylinder. The testing in this chapter took place across a period of approximately three weeks in late 2023. It involved several different approaches for measuring the thermal characteristics of the cylinder. This was done with the intention of validating models of hot water heating to be tested in later stages of this research.

4.2 Overall Aims of Testing

This part of the research has several components the author believes are important to current and future work on the control of hot water cylinders. Firstly, given the large size and potential dangers associated with hot water cylinders, few people (except for certain professionals such as plumbers) have direct experience with the details of connecting and managing an active hot water supply. The author feels the experience

of setting up and experimenting such system has much value in developing intuition to inform practical research carried out in the future.

Related to this is the question of how best to measure the physical properties of the hot water system for accurate modelling and simulation. The accuracy of many single cylinder and multi-system simulations in the literature take for granted that the underlying hot water models are accurate. Without validation using measured data, it is difficult to assess the results of the work presented, which detracts from the authority of the research and its conclusions.

Further to this, as non-linear systems, if the detailed behaviours of water in the heated water column (such as stratification) are to be well understood, the theoretical products of the model must be checked against the physical reality provided by measurements of the system's actual performance.

By measuring the temperature of water at differing heights in the water column, the work done in this section aims to achieve just such a matching between theory and real-world behaviour, to form the basis of an accurate model of hot water cylinder behaviour.

A key aim of the work of Jack et al. in [70] was to develop a simple model of a hot water cylinder that required minimal site-specific data to work with 'reasonable accuracy' [70]. 'Site-specific' data like this might include detailed parameters unique to individual hot water cylinders that would otherwise require expensive and time-consuming system identification techniques to learn.

Such an approach would make it impractical to evaluate a large number of hot water cylinders in the event that large scale DER is rolled out to target hot water storage.

The models proposed by Jack [70], and Ritchie [59], require only knowledge of a tank's volume, surface area, heating element power and tap flow rate. These are all parameters that could be read off the nameplate of a standard hot water cylinder with little effort, avoiding the complexity of other approaches.

A final parameter required by the authors, and one not readily available from equipment manufacturers, is the thermal resistance of the tank. This is a difficult

parameter to obtain, as it averages the effects of heat loss due to conduction, convection and radiation into a single value, across the entire geometry and physical composition of the various components of the system. The estimation of this value will occupy the following sections of this chapter.

4.3 Experimental Set-up

The tests and measurements in this investigation were carried out on a 180 litre, Thermann electric hot water cylinder. Thermann are an Australian owned and manufactured brand of hot water cylinder manufacturers. They have been operating in the Australian and New Zealand markets since 2012[15].

The specifications of the N180TBH124 cylinder are listed in Table 6 for clarity, but are also included in more detail in the appendix:

Table 6. Therman hot water cylinder nameplate data.

Brand:	Thermann
Model:	N180THB124
Volume:	176 L
Height:	1.78 m
Diameter:	0.492 m
Heating element:	2.4 kW

The shape and dimensions of the N180THMB124 cylinder are shown in the schematic of Fig. 20. The tank is a vertically oriented type, with a single heating element positioned close to its base and has an H/d ratio of 3.6, within the range required for good stratification [119]. The internal cylinder is of vitreous enamel coated steel construction to reduce corrosion and the buildup of bacteria that can lead to the spread of water born organisms such as cause legionnaires disease [120].

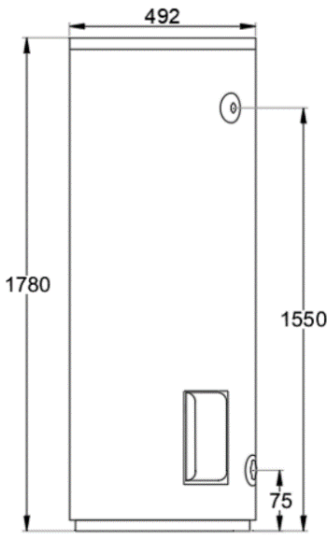


Fig. 20. Thermann N180THMB124 schematic.

The outer jacket of the cylinder is also made of a thin sheet steel. Between the inner and outer layers is a 50 mm thick

[15] https://www.thermann.com.au/about/

layer of polyurethane foam, as is typical of most ‘better’ quality hot water cylinders available in New Zealand [20].

Of note for this model, is a notice of non-compliance with AS/NZS 4692.2:2005 dating from 2018 to 2019 from EECA, regarding insulation of an external component in the cylinder. This is matter is discussed in more detail in section 4.9.

The basic arrangement of equipment to test the performance and characteristics of the tank are illustrated in the single line diagram of Fig. 21, but is also reproduced in full in appendix B.

The system includes a single hot water cylinder connected to the mains water supply. The main supply is connected to several devices to measure and log water flow, power consumption and temperature for the duration of the experiments.

Water flow into the tank was measured by an analogue Huba flow sensor plumbed into the hot water cylinder inlet pipe (Appendix C). A second flow sensor was attached to the outlet pipe.

An ordinary shower handset attached to the outflow, and a ball valve fitted to the outflow pipe were used to regulate the volume of water flow through the cylinder. Back pressure from these limited the overall flow to within 5-10 L/min, in keeping with reasonable flow rates for New Zealand houses [20].

A hand adjustable ball valve, however, allowed the flow rate to be further refined, and set to a pre-determined value, which for this experiment as set at 10 L/min [70].

The ratio of hot and cold water mixing from the outlet of the cylinder to the shower head was regulated by a mixing valve. The setting of the mixing valve was adjusted to maintain the water at the shower head below 50 ˚C.

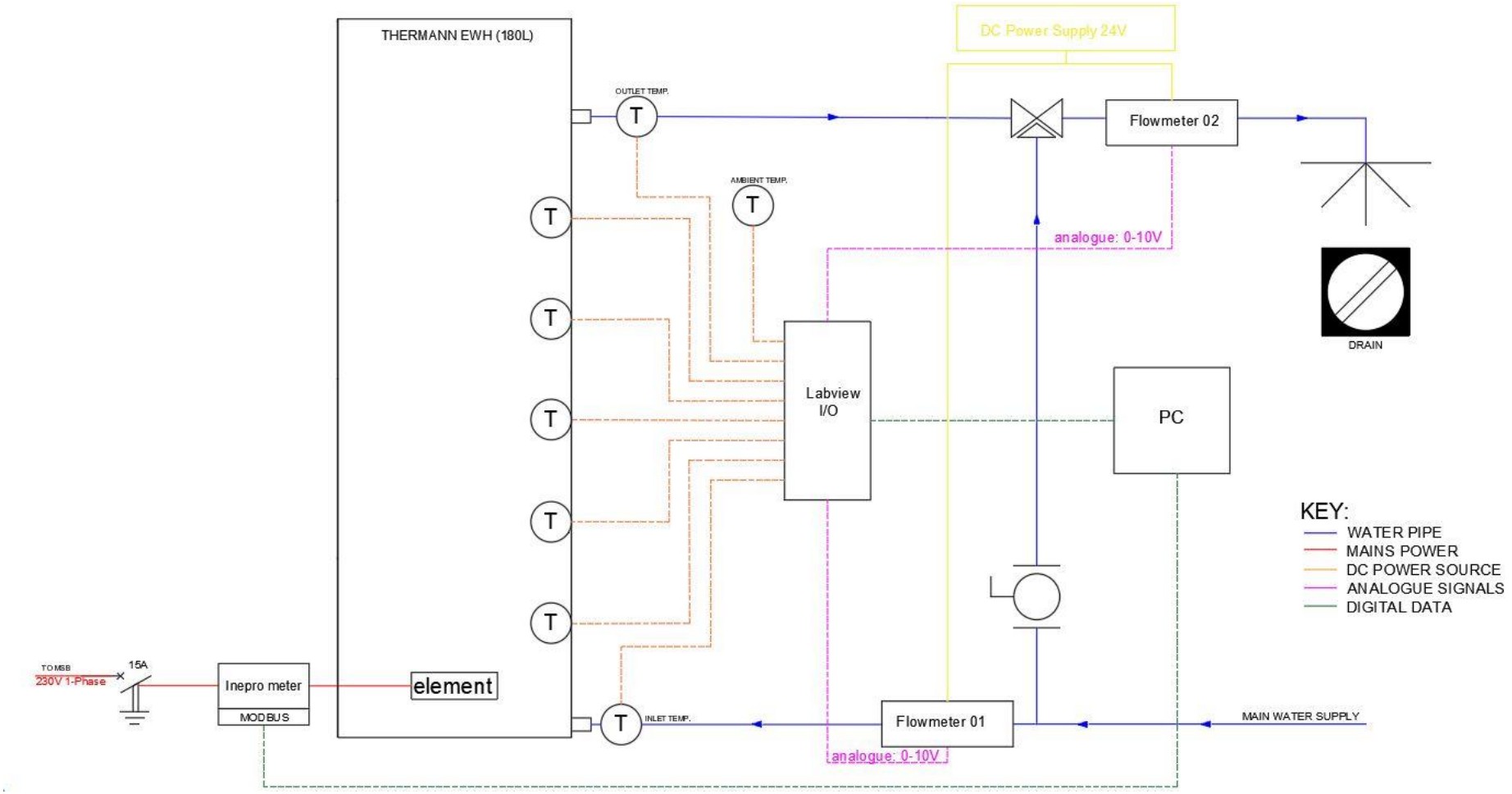


Fig. 21. Experimental setup schematic drawing.

4.4 Thermocouple Placement

Temperature data was acquired through eight T-type thermocouples attached to the wall of the inner chamber of the hot water cylinder.

Fig. 22. Insulation core removed for thermocouple placement. The wall of the inside cylinder is visible.

T-type thermocouples were used in the experiment partly because of their accuracy and availability. Formed by the pairing of Copper and Constantam (55% Cu/ 45% Ni) alloy, the two dissimilar metals work to produce a small voltage difference at their junction dependent on temperature [121]. T-type thermocouples are well suited to laboratory applications, and suitable for use in environments between 70 K to 620 K. In this range, T-type thermocouples will typically have a range of +/- 1 ˚C [121].

Six thermocouples were spaced equidistant from each other along a line running from the base of the tank to the level of the water outlet. The spacing distance between thermocouples was approximately 250 mm. One thermocouple was placed on the water inlet pipe and another used to measure ambient air temperature. Eight thermocouples in total were deployed.

With this spacing, thermocouple T01 was positioned approximately 280 mm above the height of the heating element in the water column. This was done to prevent the temperature of this thermocouple being too sensitive to the state of the heating element.

To reach the wall of the internal cylinder and place the thermocouple, a hole saw was used to remove the top layer of metal on the wall of the outer steel jacket. This allowed for the polystyrene foam around the intended position of each thermocouple to be removed and expose the inner cylinder wall (Fig. 22).

Fig. 23. Hot water cylinder setup showing sealed thermocouple plugs.

An alternative to this approach was instead to insert thermocouple pockets. The pockets would penetrate the inner

cylinder wall, allowing the thermocouples to be inserted directly into the water column. Doing so however, required the pockets to be welded to the thin wall of the tank using specialist welding equipment and expertise that was not available.

Instead, the attachment of the thermocouples to the outside of the inner tank wall was a compromise, trading off accuracy with time and cost.

Placing copper plates at the end of each plug, the tip of the thermocouple was sandwiched hard up against the cylinder wall between the copper and polystyrene plug. Thermal paste facilitated the conduction of heat from the heated inner cylinder wall to the through to the thermocouple.

Finally, each polystyrene core was reinserted as a plug and taped in place to seal each core and reduce the loss of heat from the tank due to gaps in the insulating foam and metal case, thus preserving the integrity of the insulating polyurethane foam insulation.

Doing so also helped ensure the tip of the thermocouple was firmly pressed against the cylinder wall, thereby minimising the contact resistance between the thermocouple tip and the inner hot water cylinder wall [122].

4.5 Data Acquisition

Three methods of data acquisition were used: a temperature logger for interpreting thermocouple voltage measurements; a power meter for logging energy flow into the heating element; and Analogue to Digital Conversion (ADC) for interpreting analogue voltage signals from the water flow meters.

4.5.1 Temperature Data Acquisition

Temperature data from the thermocouples was collected by a Pico TC-08 Temperature logger. Thermocouples work according to an electromagnetic phenomenon known as the Seeback effect, in which two wires of dissimilar metals connected to each other at a point known as the ‘measuring junction’ [123]. The voltage difference registered at the measuring junction is used to infer the environmental temperature, using a known profile for the pair of dissimilar metals used in the thermocouple.

The Pico TC-08 thermocouple datalogger[16] has the added benefit of built in Cold Junction Compensation (CJC), to prevent temperature drift at the junction as the ambient temperature shifts diurnally through the course of the day.

4.5.2 Power Data Acquisition

Running alongside the Pico TC-08 was an Inepro power meter connected via a Modbus RS485 to USB to interface with a desktop instance of Labview.

The Inepro Pro-1[17] is DIN rail mounted power meter capable of measuring voltage, current, power and several other variables of power flow. In this case, it was wired in series with the heating element of the hot water cylinder.

4.5.3 Flow Meter Data Acquisition

The two flow meters connected to the inlet and outlet pipes generated a 0-10 V analogue signal depending on the rate of water flow. This voltage was converted into a digital signal using the National Instruments NI2209 Data Acquisition (DAQ)[18].

The NI2209 is also capable of DAQ for thermocouple probes, however, an early attempt to read temperature data into Labview from this device was abandoned, because this model lacks CJC functionality. Consequently, temperatures measurements were found to be susceptible to temperature drift in line with changing ambient temperature of the surrounding environment.

4.6 Theoretical Estimation of Thermal Resistance

An initial estimation of the thermal conductance of the hot water cylinder is made here, based on a 1-D lumped analysis of the physical layers between the centre of the interior cylinder and the layer of air in direct contact with the outer shell.

The basic structure of modern electric hot water cylinders is a smaller cylinder sitting inside a larger outer cylinder made of a corrosion resistant galvanized steel shell. The space between the two is filled with an insulating material [20]. Historically in New Zealand, this was often been

[16] https://www.picotech.com/data-logger/tc-08/thermocouple-data-logger
[17] https://ineprometering.com/smart-pro-series/
[18] https://www.ni.com/en/shop/compactdaq.html

constructed using lower quality materials such as fiberglass, but in recent years, with more stringent standards, is more likely to be a polyurethane foam [20].

4.6.1 Mechanisms of Heat Loss and 1-D Analysis

There are three basic mechanisms for heat loss from physical objects to the external environment: conduction; convection; and radiative heat loss [122].

In this evaluation of heat loss from the hot water cylinder under test, the effects of radiative loss are assumed to be minimal, given the thick insulating layer and small difference in temperature observed between the outer shell and ambient air temperature.

Instead, the bulk of the heat loss is assumed to be through conduction of heat laterally from the interior of the hot water cylinder to its exterior, and heat loss from the surface of the exterior shell of the hot water cylinder, due to convection with air in the external environment.

At the convective stages of heat flow, the transfer of heat from the hard wall of the outer hot water cylinder shell to the fluid (air) moving across it is provided by Newton's law of cooling for convection [122]:

$$q" = h(T_s - T_\infty) \qquad (4.1)$$

Where:

q" = convective heat flux;

T_s = temperature of the hot water cylinder outer jacket surface;

T_∞ = temperature of the air in the environment of the hot water cylinder; and

h = convection heat transfer coefficient (K/W).

For the region between the walls of the inner hot water cylinder container and the skin of the outer jacket, where a thick layer of insulating polyurethane foam is installed, the flow of heat is assumed to be mainly conductive.

For the case of conductive heat transfer, from Fourier's heat transfer equation we have [122]:

$$q" = -k\frac{\Delta T}{L} \qquad (4.2)$$

In this case:
- q” = conductive heat flux per unit area;
- ΔT = temperature difference between the mean hot water temperature and the ambient environment;
- k = thermal conductivity of the material separating the inner and outer surfaces of the container (K/W); and
- L = thickness of the material (m).

Heat lost through both conduction and convection are related to thermal conductivity by the overall heat transfer coefficient, U. This relationship is governed by the equation:

$$Q = UA \cdot \Delta T \qquad (4.3)$$

Where the heat transfer coefficient U is the reciprocal of the thermal resistance of the material through which heat passes [124].

Equating the terms in expression 4.1 with 4.2 through 4.3 yields:

$$R_{val} = \frac{1}{UA} \qquad (4.4)$$

The thermal resistance of a material is indicative of how much resistance there is to the flow of heat through a body per unit area. Calculation of individual R-values in a composite material allows the overall thermal resistance R_{val} to be calculated by a method analogous to the addition of resistances elements in an electrical circuit [122].

If each layer of the composite material has known thermal conductivity and thickness, a simplified 1-D model of heat flow can be developed, allowing heat loss from convection and conduction to be represented with a single value represented by R_{val}.

To relate this back to the hot water cylinder to be tested in this evaluation, heat lost from the water column in the middle of the tank to the surrounding environment must pass by conduction through the metal wall of the inner cylinder to the insulating polyurethane layer between cylinders and the outer steel cylinder jacket. It is then transferred to the surrounding environment

by convection, as air flows past the cylinder and absorbs thermal energy.

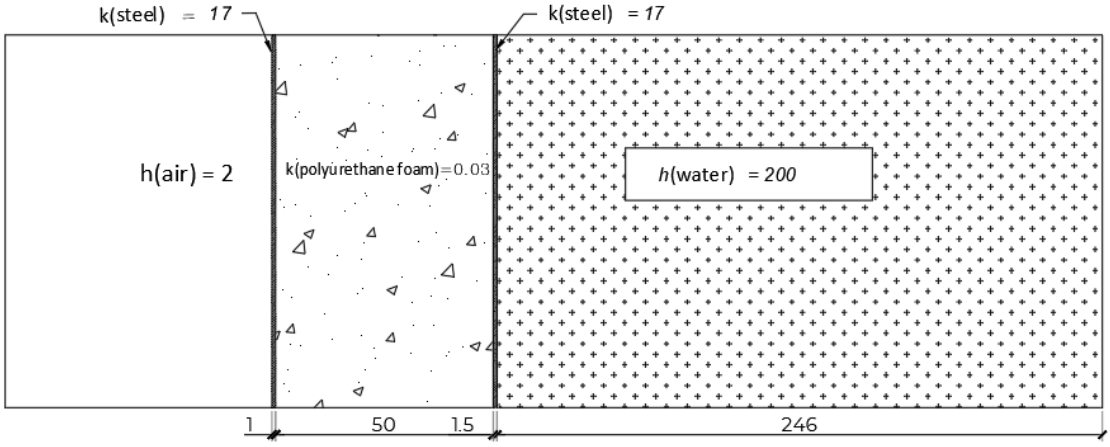


Fig. 24. Insulating layers of the hot water cylinder.

Given the cylindrical shape of the container, the curvature of the container wall causes heat to be spread over an increasingly wide area as it moves outwards from the centre of the cylinder to the cylinder walls. The R-value on the curved surface of the hot water cylinder walls of radius *r*, can be calculated according to [125]:

$$R = \frac{r_2 \ln\left(\frac{r_2}{r_1}\right)}{k} \qquad (4.5)$$

Each of these has its own characteristic R-values, which are added consecutively as lumped thermal resistance elements to sum to a final estimate of the value R_{total} for the side wall of the cylinder.

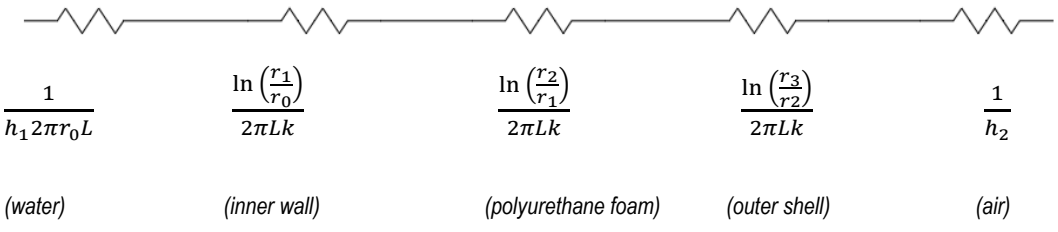


Fig. 25. 1-D equivalent circuit using lumped elements.

Since the area of the side wall is approximately 2.75 m^2, an estimate of the equivalent R-value of the composite layers is found by multiplying R by the area to yield the estimation: $R_{val} = 1.93$ K/W.

Calculation for the top surface of the cylinder, with slightly thicker insulation (L=0.06 m), yielded a thermal resistance R_{val} = 2.06 K/W.

Table 7. Simplified calculation of R for the vertical tank wall. (Values of k sourced from [125].)

Layer type	Thermal conductivity k (W/mK)	r (m)	Thermal Resistance (m^2K/W)
Water (internal)	20	0.193	0.0023
Inner cylinder wall	17	0.1945	4.07e-5
Insulating layer	0.03	0.2445	0.6819
Outer steel jacket	17	0.0.246	3.22e-5
Air (horizontal)	200	-	0.0182
Total R [m^2K/W]			0.7024

This is a speculative number, since the analysis is along a single dimension, and assumes a steady state environment. Neither is an estimate made for the thermal R-value of the bottom layer of the cylinder, since this was not accessible and could not be measured.

The value does however fall within the range R = 1-3 stated by Jack in [70] as typical of hot water cylinders in New Zealand, and is useful as a sense check for estimating the thermal resistance of a working hot water cylinder in an experimental setting. The approach is also consistent with that followed by other researchers. For example, in [126], it is noted that a common practice is to calculate U-values using 'simple one dimensional analysis for heat loss through the sides of the storage tank', where the assumption is made that the 'U-value is similar on all sides'. Since R and U are inversely related, the same reasoning applies.

4.7 Evaluation of Cylinder Characteristics

To go beyond the simple 1-D case, and attempt to calculate the thermal resistance of a hot water cylinder experimentally, one approach taken was based on that of Cloete, who documented his experiments comparing the performances of horizontal and vertical hot water cylinders in [127] . As in that work, after testing the accuracy of the thermocouples given their location on the

exterior of the hot water cylinder, the primary aim was to measure the thermal resistance of the hot water cylinder across its full surface area. Unlike the approach in [127] which also estimated the thermal resistance of a horizontally mounted cylinder, this testing deal with a single vertically standing tank.

4.7.1 Accuracy of Thermocouple Measurements

In the first stage of testing, regression analysis was used to estimate the difference between the true internal temperature of the top layer of hot water in the tank, and the temperature measured at the outlet by thermocouple T06.

Assuming the internal temperature of the hot water cylinder is likely to be slightly offset from the ambient temperature of the external environment, the degree of offset was estimated by a transfer function of thermocouple measurements, fitted to outlet temperature readings, as carried out by Cloete in [127].

The degree of this offset is estimated according to the transfer function:

$$T_{cylinder} = (T_{out} - T_{ambient})\beta + \alpha \qquad (4.6)$$

From these measurements, an estimation of permanent offset between the thermostat setpoint temperature and cylinder outlet was determined according to the value of α. The slope of this curve allowed for estimation of β.

To determine α and β, the hot water cylinder was flushed of standing water to remove as much heat as possible from the cylinder and ensure the entire volume of water in the tank was uniformly equal in temperature to the inlet. After this, outlet water temperature readings were captured at roughly five-minute intervals, by running approximately 250 ml of water flow into a cup at the outlet pipe for each measurement. The temperature of the top layer of this sample was then measured using a hand-held thermocouple with a digital display.

The collected data is presented as a scatter plot in Fig. 26. The trend line shows that at lower temperatures, there is an absolute error in measurement of approximately 0.5 ^{o}C. Since the value

of alpha is found to be slightly less than one, estimations of temperature at the water outlet are slightly underestimated.

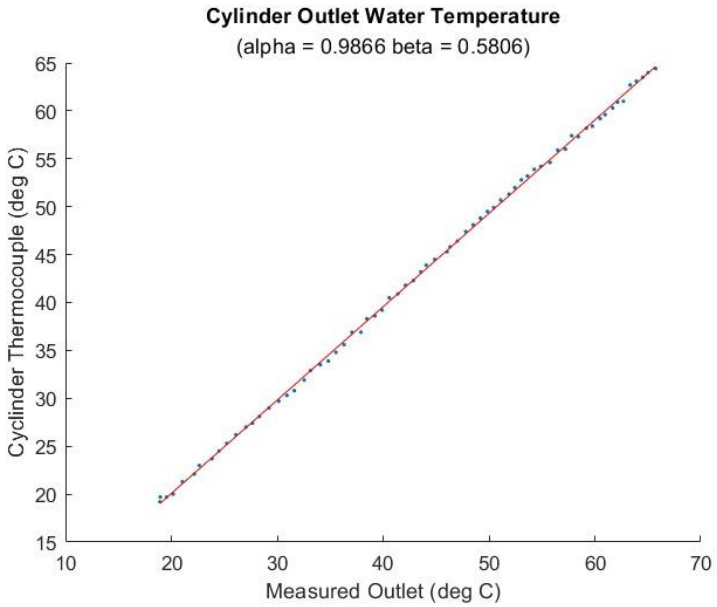


Fig. 26. Correspondence of measured thermocouple temperature to direct measurements of water temperature at shower head outlet.

The results of this test demonstrate that the temperatures measured by thermocouples on the wall of the hot water cylinder are a good proxy for the actual temperature of water in the water column at the same heights.

The apparent simplicity of this procedure, however, does not allow for several effects within the water column that are hidden from view as heat moves by convection through the tank.

For example, the effect of cooling at the tank wall is an important factor in the movement of water that was discussed in [126] and is also relevant here. The complexity of water heating around the heating element also causes heat to rise unevenly from the base of the tank. At the walls of the tank, heat is lost to conduction through the wall of the cylinder. The cooler water in contact with the cylinder wall is relatively denser and sinks. For this reason, it was expected that the temperatures measured by the thermocouples would be noticeably cooler than water close to the centre of the cylinder. However, on the basis of Cruickshank et al.'s observations, the change in

temperature in the horizontal plane across the breadth of the cylinder is still small (≈0.02 ˚C) [126], and was neglected.

4.7.2 Heating Test

In the next stage of the experiment, the tank was flushed with cold water from the inlet pipe, by opening the ball valve at the outlet and emptying 200 L of cold water. At the end of this process, the water in the cylinder was of uniform temperature (17 ˚C).

Starting at time zero, power to the heating element was activated, and the tank began to heat uniformly until after approximately six hours, the temperature of water in the tank reached 70 ˚C and the thermostat disconnected the heating element from the power supply.

The time-series thermocouple temperature measurements for this test are shown in Fig. 27.

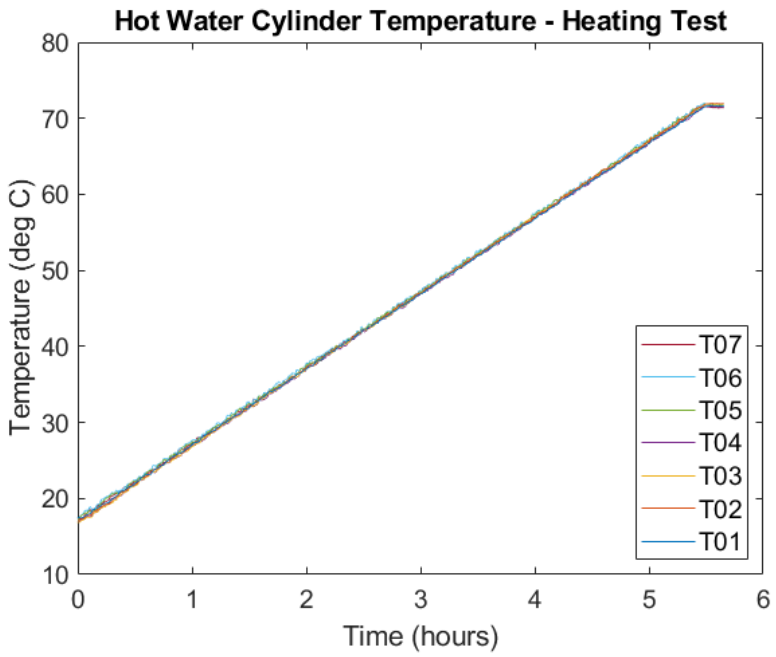


Fig. 27. Heating rate of hot water cylinder from ambient room temperature.

Of note, the heating of the water at all levels of the tank is uniform, and the temperature measured at the bottom thermocouple (T01) rises at approximately the same rate as the thermocouples in the layers above, all the way to the top thermocouple, T07. During the constant heating phase, where there is no disruption to the water column from outside disturbance, no evidence of stratification is seen and the entire water column appears to heats as a single body.

The apparent linearity of the heating is a consequence of the insulation of the tank and the high heat of the heating element relative to the water around it. There is minimal heat loss to the environment. Closer examination of the heating curve reveals some linearity in the early stages of heating, but a degree of curvature in the curve as heating progresses.

4.8 Estimating the Overall Thermal Resistance

In this section, several different approaches used by other researchers are used to find a value for the overall thermal resistance of the hot water cylinder.

4.8.1 Static Cooldown

The first test involved heating the tank up to the set-point temperature, turning the element off, and then allowing the water inside the tank to cool passively to the ambient temperature.

The temperature profiles captured from each thermocouple are plotted in Fig. 28.

The temperatures measured at each thermocouple decay at slightly different rates, suggesting the flow of heat out of the tank is not uniform through the tank. The gradient of curve is also steeper in the first days, where the temperature differential between the water and ambient air temperature in the constant temperature laboratory environment is greatest.

Furthermore, the rate of decay is higher at lower levels in the tank. The thermocouple for Sensor 1 deviates at a rate much faster than the sensor above it, and is similar to the cooling behaviour observed by the authors of [126]. In that study, there was some suspicion that the layer of insulation on the bottom of the hot water cylinder was thinner than that used for the side walls and top of the tank. The relatively lower temperature of the concrete pad (17 $^{\circ}$C) relative to the average ambient air temperature (23 $^{\circ}$C) may also have contributed to this effect.

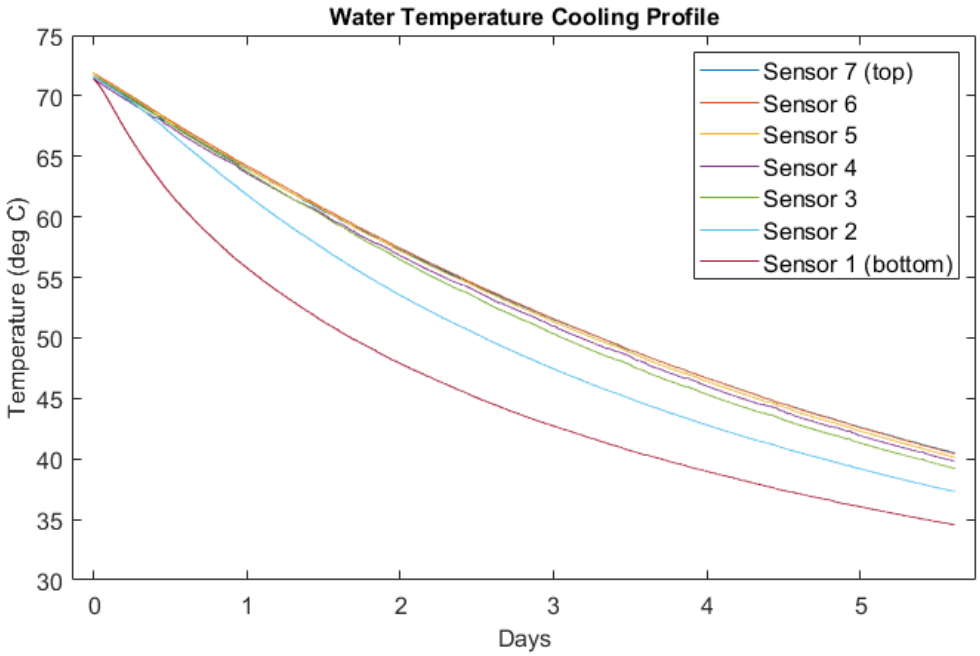


Fig. 28. Cooling profile measured over five days.

Interestingly, heat loss is also seen at the level of the second thermocouple which, as well as losing heat to the colder layer below it, may also be seeing heat leaked from the inlet pipe connecting the cold-water source to the base of the hot water cylinder. Some effort was made to put extra foam insulation around this area, but in some sections, particularly at corners and sharp bends, exposed sections of copper are notable and likely to have contributed to the extra gradient observed in the energy loss data at the bottom two nodes.

4.8.2 Estimation of Thermal Resistance - Method 1

The overarching ordinary differential equation for a hot water cylinder, is based on the equation for fully mixed single node hot water cylinder, to be derived in chapter 4:

$$C_{\upsilon}\rho V\frac{dT(t)}{dt} = UA[T(t) - T_o(t)] \qquad (4.6)$$

Where:

Cυ= heat capacity of water (4180 J/kg·K);
ρ = density of water (992 kg/m^3);
V= Volume of tank (m^3);
F(t) = volumetric flow rate of water (m^3/min); and
U = Thermal transmittance through the surface of tank (W/m^{2}K).

This suggests a linear relationship between the natural logarithms of $\frac{dT(t)}{dt}$ and $[T(t) - T_o(t)]$, with slope $\frac{UA}{C_v \rho V}$.

The MATLAB function 'fitlm' was used to fit a linear regression of measured values of temperature in the time series data. The resulting plots are drawn in Fig. 29, starting with the upper layer and working down to the bottom of the tank. The values of U and R, estimated from the slope of these lines are presented in Table 8.

Based on this approach, the thermal resistance of the hot water cylinder as a single lumped element is taken as the average of the calculated values at differing heights in the tank's height.

The estimated value of overall thermal resistance using this first approach is approximately R_{val} = 1.74 K/W.

Table 8. Estimation of R from measures of gradient plotted in Fig. 29.

Thermocouple Node	**UA(W/m^{2}K)**	**U(W/K)**	**R (K/W)**
Th06	1.92	0.63	1.58
Th05	1.89	0.62	1.6
Th04	1.88	0.62	1.62
Th03	1.79	0.59	1.69
Th02	1.61	0.53	1.87
Th01	1.48	0.47	2.05
Average:	1.75	0.49	1.74

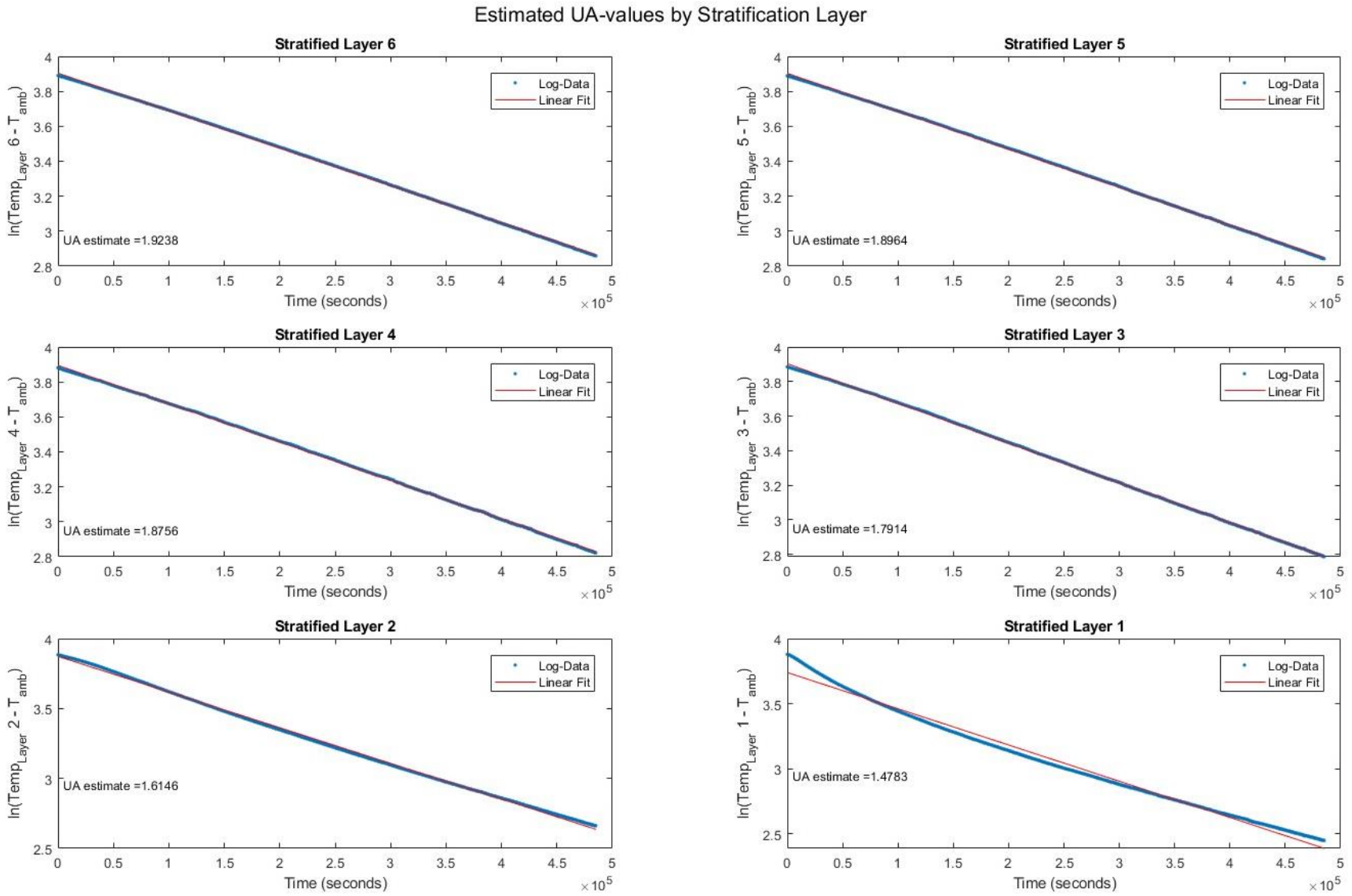


Fig. 29. Estimations of thermal conductivity from line gradient.

4.8.3 Estimation of Thermal Resistance - Method 2

In [126], Cruickshank et al. carry out a similar analysis the one presented above, but instead focus their attention on the calculation of thermal conductivity U, the inverse of thermal resistance. This method also differs in that, each node layer is treated separately, and the value used to calculate U is based on the rate of heat loss between time steps.

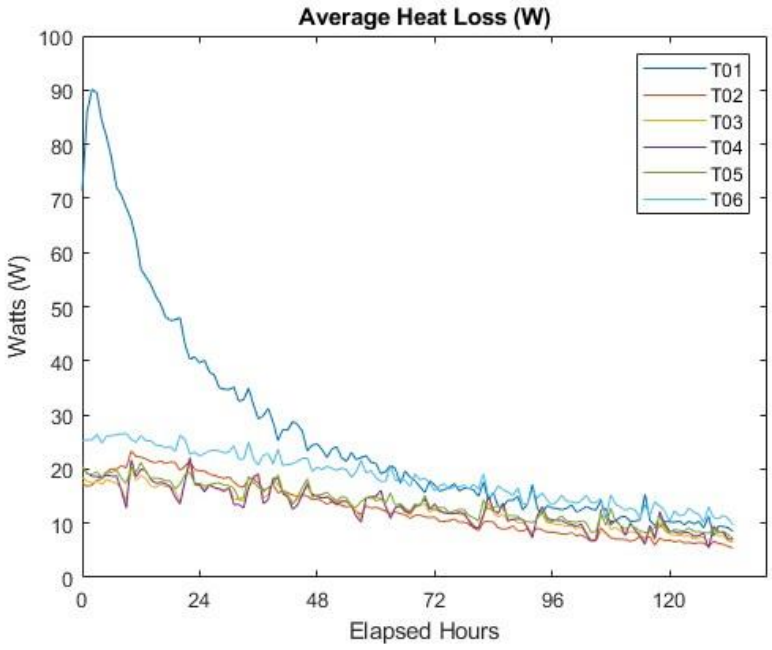


Fig. 30. Average heat loss rate during cooling.

The greatest rates of heat loss are measured during the early hours of cooling, when the temperature difference between the heated water in the tank and the outside environment is greatest. The steepest gradient is observed at the in the lowest node, but after approximately 100 hours, the heat loss rates converge for all stratified layers, indicating full mixing has occurred and energy is spread evenly through the tank.

The averaged results for each node layer are tabulated in Table 9. Following the method described in [126], the heat transfer rate (ΔQ) for each node layer was measured at each time step of the testing period.

Given that the cooling down test was conducted over many days, a one-hour time step was chosen to reduce the noise in the resulting plot of average heat loss, as shown in Fig. 30.

The ΔQ values were then divided by the product of the node's surface area and the temperature difference between the node and the ambient air.

The mean values derived from these steps are recorded as the area weighted U-value recorded in Table 9. The corresponding R-value for each node was calculated as the inverse of the product of the U-value and node's surface area:

Table 9. Estimated values of thermal conductance and thermal resistance based on rates of heat loss at different heights in the water column.

	Surface area (m^2)	**Average heat loss rate (W)**	**Area weighted average U-value (W/m^2K)**	**R-value equivalent (K/W)**
Node 1	0.87	26.4	1.12	1.02
Node 2	0.35	12.6	1.24	2.3
Node 3	0.37	12.7	1.11	2.4
Node 4	0.39	13.0	1.08	2.4
Node 5	0.41	13.6	1.07	2.3
Node 6	0.74	18.4	0.78	1.7
Mean	-	13.8	0.91	2.02

4.8.4 Estimation of R using Standing Losses – Method 3

In this test, the cylinder was heated to a set-point temperature and kept there for the duration of the test. As no heated water was drawn during this time, the only heat able to leave the system is assumed to be as a thermal loss through the walls of the hot water cylinder.

The rate of thermal loss is calculated from:

$$\dot{Q}_{loss} = \frac{1}{R}\left(T_{cyl}(t) - T_{amb}(t)\right) \quad (4.10)$$

Knowing also that:

$$Q_{loss} = Q_{elec} = \int_0^N P_{elec}(t)dt \quad (4.11)$$

Equating the two and rearranging to make R the subject, allows the thermal resistance of the cylinder to be estimated as [5]:

$$R = \frac{(\hat{T}_{cyl} - \hat{T}_{amb})}{\hat{P}_{elec}} \quad (4.12)$$

Where:

$\hat{T}_{cyl}$ = estimate of the temperature base on the transfer function in (4.6); and

$\hat{P}_{elec}$ = the average power draw of the heating element over the period of testing.

In this experiment, heating data was collected for a total of 7 days and is plotted in Fig. 30 showing the temperature response at each thermocouple (top) relative to the state of the heating element (bottom), under thermostat control.

In the lower plot of Fig. 32, the state of the heating elements over the length of the test period are also plotted, showing the element in either ‘on’ or ‘off’ states depending on the water temperature measured by the hot water cylinder thermostat.

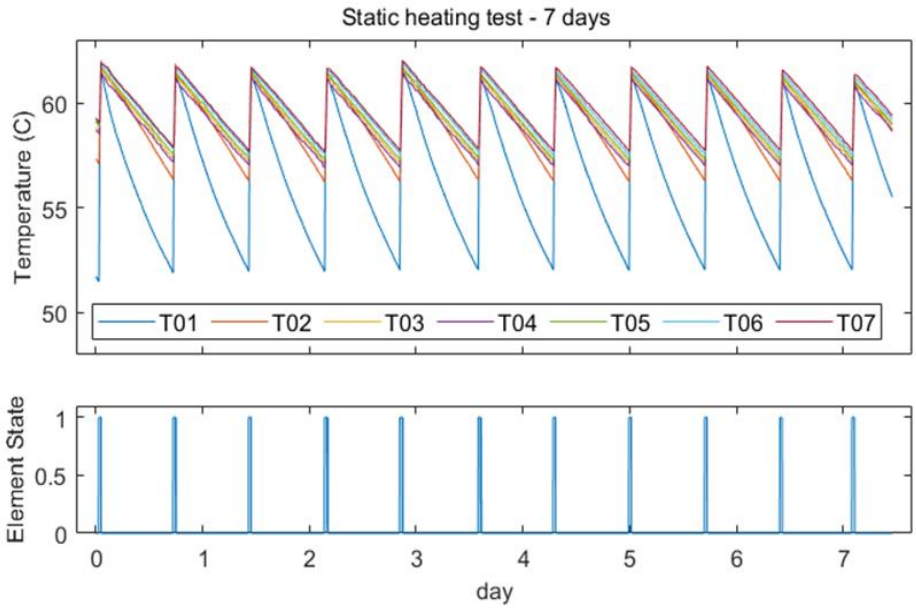


Figure 31. Total energy consumption during the static heating test over seven days. The element heating periods are uniform, but there is variability across days, due to minor fluctuations in the ambient environment.

The power used to heat the water was collected by the Inepro Pro 1 power meter connected through the mains power supply to the heating element of the hot water cylinder. In this experiment, the thermostat temperature dial was set to 60 ^{O}C, but given the nature of the mechanism, the actual setting could only be very approximate.

The deviation from the set-point temperature in the heating case was around 1.6 ^{O}C. The temperature below the set-point temperature was similarly found to be 2.3 ^{O}C, suggesting a hysteresis of around 2 ^{O}C from the set-point temperature.

The accumulated total of energy consumed for each day is shown in

Table *10*. On days 0, 2 and 5, the heating element was triggered before on more than one occasion during the day, giving these days slightly higher energy consumption. For this reason, the individual values measured are averaged across the time of the entire seven-day testing period to derive an average daily energy consumption.

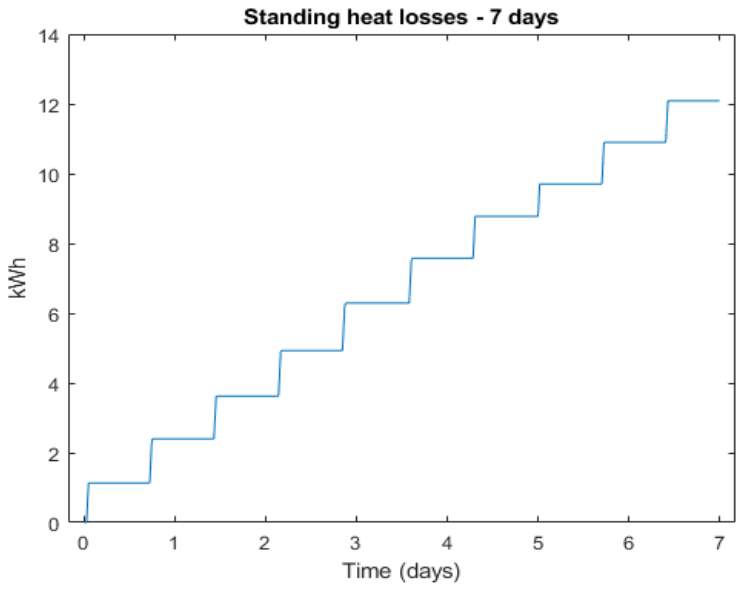


Fig. 32. Total energy consumption during the static heating test over seven days. The element heating periods are uniform, but there is variability across days, due to minor fluctuations in the ambient environment.

In this test, at T_{outlet} = 60 ^{o}C, the total electrical energy consumed over the testing period was recorded as 12.09 kWh. The average daily energy consumption was measured derived from this value is 1.72 kWh/day. Because of limitations in the time available to carry out testing, further experiments at other temperatures were not able to be made but should be carried out in the future in a more detailed analysis.

Table 10. Daily energy consumption for static heating with no hot water draw.

Day	Energy (kWh/day)	Average power (W)	U-value (W/ m^{2}K)	R_{tot} (m^{2}K/W)
0	2.39	99.9	2.6	0.39
1	1.23	51.0	1.3	0.76
2	2.67	111.1	2.9	0.35
3	1.29	53.7	1.4	0.72
4	1.2	63.7	1.7	0.61
5	2.12	88.4	2.3	0.44
6	1.19	49.6	1.3	0.78
Mean	1.72	73.9	1.92	0.57

Overall, the values estimated by this method agree with those of the cooling down scenarios. In this case, when taken over the full seven-day period, the maximum value of U recorded was between 1.3 and 2.6, the difference due to the timings of element activity where on three days out of seven, the heating element was triggered twice.

On average, over the entire seven-day period, the thermal resistance per unit area was found to be 0.57 m^2K/W, which corresponds to R-value = 1.72 K/W (2s.f.).

4.8.5 Overall Estimation of R

The overall results of the three different approaches to the calculation of the overall thermal resistance R of the hot water cylinder are presented in Table 11.

To arrive at an overall estimation, the values have been averaged to achieve the final estimation of R = 1.8 K/W. This value is lower than expected, given the initial estimation arrived at using the simple 1-D analysis of section 4.6, but is feasible given the observations of heat loss measured in 4.8.2 and commented on in section 4.9.

Table 11. Estimates of overall thermal resistance.

Testing method	**Estimated value (K/W)**
Method 1	1.7
Method 2	2.0
Method 3	1.72
Average:	1.8 (K/W)

Further testing should be carried out to arrive at more precise estimates. This could also be done using rigorous steps laid out in the procedures described in [119].
To compare the results of this testing to the estimation of R, a solution of the first order ordinary differential equation of the hot water cylinder found using MATLAB's

ODE45 solver[19], is compared to the average temperature of the hot water cylinder measured in Fig. 28.

By solving for several values of the parameter R, MATLAB's fminsearch[20] function was then used to minimise the difference between the values in the measured and estimated curve solutions, to find the closest fit between the curves. This was found to be at R = 1.7068, close to the lower end of the estimates provided, especially method 1.

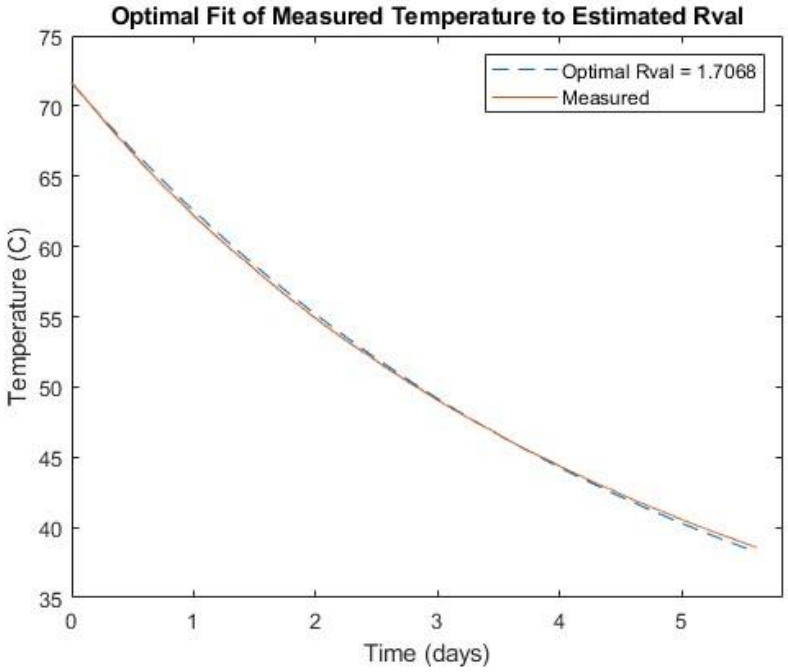


Fig. 33. Comparison of measured data to ODE45 solution of single node equation of hot water cylinder, shows good agreement with experimental estimates of overall thermal resistance.

4.9 Energy Loss Standard Comparison

The Therman N180THB124 hot water cylinder under test was donated to Auckland University of Technology for research purposes; the model having already been withdrawn from the New Zealand market.

[19] Ode45 is a popular general purpose ODE solver MATLAB that uses Range-Kutta methods to solve most general ordinary differential equations.
[20] fminsearch is an optimisation algorithm native to MATLAB, used to find the smallest solution to a set of ODE equations.

Research into the background of this model shows it had been listed in the International Accreditation New Zealand (IANZ) 'Check Testing results for 2018/2019 (Appendix C).

The certification, carried out by an IANZ accredited laboratory, tested the hot water cylinder against the Minimum Energy Performance Standards (MEPS) required by AS/NZS 4692.2:2005.

The model is listed as having failed the minimum energy performance test (MEPS) with non-compliance because of lack of insulation around the temperature/pressure relief (TPR) valve.[21]

The general requirement of AS/NZS 4692.1:2005 and AS/NZS 4692.2:2005 is for cylinders to meet a minimum standing loss performance efficiency of less than 1.6 kWh of heating energy every 24 hours with no water drawn off [129]. This includes stipulation that the cylinder also maintain its standing temperature above 55.6 ^{o} C relative to the ambient temperature.

For a cylinders of capacity greater than 90 L, a more detailed heat loss calculation is applied. In this case, the Thermann model under test has rated volume of 176 L. According to the standard, the maximum heat loss rate should be calculated as [125]:

$$\begin{aligned} \text{Maximum heat loss} &\leq 0.0048 \times V + 0.72 \\ &\leq 0.0048*176 + 0.72 \\ &\leq 1.56 \text{ kWh/day} \end{aligned} \qquad (4.13)$$

In section 4.8.2, it was established that over the five-day period of the testing, a total of 12.09 kWh of heat was lost. The measured heat loss of 1.72 kWh/day, 10.3% below the level stipulated in the standard.

The result should not be surprising since the status of the cylinder was already known, but it does underscore how relatively small areas with poor insulation can leak large amounts of heat. For these tanks, the lack of insulation around the

[21] The TPR valve is a combined temperature/pressure relief valve that releases water from the cylinder when either temperature or pressure exceed limits preset by the manufacturer [20].

pressure valve, a component that covers only a very small percentage of the overall surface area of the cylinder, has a significant thermal bridging effect.[22]

4.10 Summary

Much of the literature surrounding large scale hot water cylinder control is very theoretical, but few studies are able to demonstrate the accuracy of their models with validation to real world data; whether obtained either in the laboratory or in the field. This is especially true in New Zealand, since most published studies use data from other jurisdictions that may not always capture the characteristics of hot water cylinders used in this this country.

In this chapter, the steps involved in testing a 176 L Thermann N180THMB124 electric hot water cylinder in a laboratory setting were described. Using temperature data acquired directly from the cylinder and surrounding environment, the characteristics of the heating and cooling behaviour of water at different layers of the water column were recorded for analysis.

From this data, the overall thermal resistance of the system was determined to be between 1.7 W/K and 2 W/K, which is a value consistent with other studies and values attributed to hot water cylinders in New Zealand. The value was still lower than expected, and this is partially attributed to a defect in the design of thermal insulation noted for this model, enabling heat to leak from the device through thermal bridging around the uninsulated TPR valve on the upper section of the hot water cylinder body.

Much of this analysis focused on determination of the overall thermal resistance of the tank, a measure of the degree of heat loss to the environment through conduction of heat through the wall of the hot water cylinder, then convection of heat from the outer shell to the surrounding air.

The determination of this value is important because, as will be shown in the following chapters, *R* is a parameter crucial to many hot water cylinder models.

[22] An improved model of the 'Thermann 180THMB124', appears on an updated list of 'New Zealand Minimum Energy Performance (MEPS) Compliant Electric Storage Water Heaters" in 2023 [124].

Establishing this value allows for more accurate characterization of several hot water models, which will form the basis of discussion in the following chapters of this research.

Chapter 5 Electric Water Heater Simulation models

Electrically heated water cylinders are a comparatively simple technology for residential energy storage compared to other technologies such as household batteries. Their apparent simplicity, however, hides some very complex natural phenomena around the heating and cooling behaviour of water at the molecular level. Though the finer details of this behaviour are often abstracted away, researchers must still be able to approximate the resulting effects. In some cases, detailed and computationally expensive models are developed to accurately predict the potential benefits of a new design. In other cases, simplified models are sufficient to allow the simultaneous simulation of large numbers of hot water cylinders across a population, to examine the potential of large-scale energy storage.

There are many ways in which researchers try to model the underlying, complex phenomena occurring inside hot water storage, and many classes of computational model have been developed to achieve these aims. In this chapter, an overview of the various category of hot water models found in the literature will be provided, after which the derivation of three representative models will be given, to show the basic assumptions made and capabilities of each.

5.1 Electric Water Heaters as Energy Storage Devices

Electric water heaters are naturally suited to the task of storing energy in residential settings, due to their inherent simplicity and reliability. In comparison to other energy storage media, water has a relatively high thermal capacity (4.2 kJ/kg·K), is available everywhere and is non-toxic [117]. Compared to alternative, more advanced forms of energy technologies, they are also cheap [130].With electric hot water cylinders present in almost nine out of ten New Zealand houses in sizes ranging from 130 L–200 L [131], Jack et al. estimate that each house in New Zealand has as much as 10 kWh of potential storage at its disposal [70], which could be of use to help stabilise the wider electrical grid.

The apparent simplicity of operation however belies a surprisingly complex interplay between water contained in the tank and the environment it interacts with, which is governed by the laws of thermodynamics and thermo-fluids [132].

5.1.1 Description of Internal Operation

A rather curious feature of water is that, as it is heated beyond its freezing point, its density increases until it reaches its maximally dense state at 4 °C [133], [134]. After this 'anomalous' temperature is achieved, adding more heat to cold water makes it expand and become more buoyant relative to cooler surrounding water [133]. It is the increased buoyancy of water as it is heated that is the main factor behind the complex dynamics that play out silently in the darkness of every residential hot water cylinder. On contact with the resistive element at the base of the cylinder, heated water expands and rises through the interior of the tank, creating convection currents that quickly spread to make contact with the outer edges of the tank wall [116], where it cools and sinks.

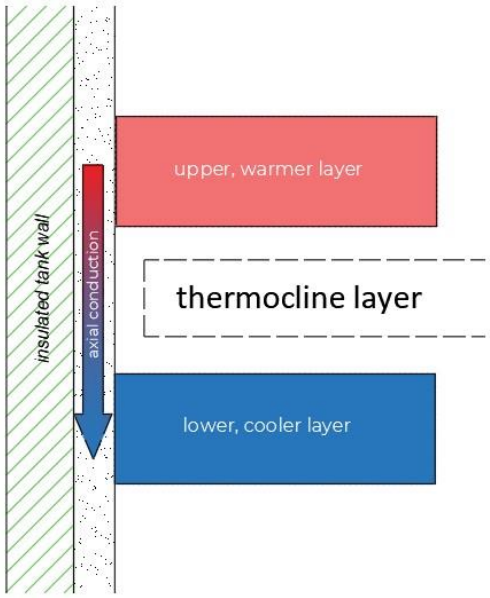


Fig. 34. Axial cooling at the tank wall. Heat is conducted at the edge of an upper layer of warm water, down through the steel wall of the cylinder to cooler layers below the thermocline. (Adapted from [110])

The cycle of heat generated buoyancy in the interior, followed by cooling and sinking along the edges create an overall 'convective drive' [117] that Chandra et al. describes as 'axial wall conduction' which drives a "natural recirculation loop inside

the tank" [117]. This process forms the temperature gradients that drive the movement of water through the water column.

But the behaviour of water, and the 'recirculation loop' described above is not the full story, occurring as it does as the boundaries of the container. There is another effect that counterbalances the convective churning, driven by plumes of heated water rising in the water column, which is 'stratification' [117], [119].

5.1.2 Stratification of Water Layers

Stratification is the tendency of distinct layers of water to form in the water column, sorted vertically in ascending order according to temperature [119]. Since the buoyancy of the water in a plume is driven by the relative degree of expansion it has undergone due to heating; cold water flowing into the bottom of a hot water cylinder will tend to stay at the bottom of the tank. In contrast, the hottest water, being more buoyant than the surrounding mass of water, collects in a layer at the top of the cylinder. In contrast, cooler layers will tend to sink slightly and spread to cover a layer floating beneath. Water cooled at the edges of the tank will sink further, and not stop until it reaches those levels of the container where the surrounding water is at the same temperature [135].

The boundary between two thermally distinct layers of water is a called the 'thermocline' [116]. This is often thought of as a very thin layer, but in actual fact can occupy a sizable proportion of the water column, depending on the relative temperature difference and effects of turbulence experienced by other layers in the tank [116].

The thermocline serves the important role of separating warm water from cold, thus reducing mixing between stratified layers and thereby improving the efficiency of the overall system [116]. In fact, the larger the separation in temperature between hotter and colder layers in the tank, the greater is overall efficiency [119].

There are several reasons for this. Cold water entering the tank, if it mixes with heated water, will reduce the temperature of the mixed water mass to the average of the hot and cold-water temperatures, requiring the heating element to turn on again to heat the water back up to its previous temperature. If instead, thermal

stratification is already present in the tank, the new water entering at the base will only mix with other layers of similarly cold water, and the temperature of the warmer layers above is preserved.

The mixing that occurs in a stratified tank is consequently less likely to affect the quality of heat energy supplied to the user due to the natural separation of hot water from cold in different layers of the water column [117].

If the hottest water settles in the upper regions of the tank, the energy it contains, being stored in a smaller volume, will also be lost across a relatively smaller surface area, reducing losses to the ambient environment, the first of four main mechanisms of heat loss in storage devices listed by Zuriga [136].

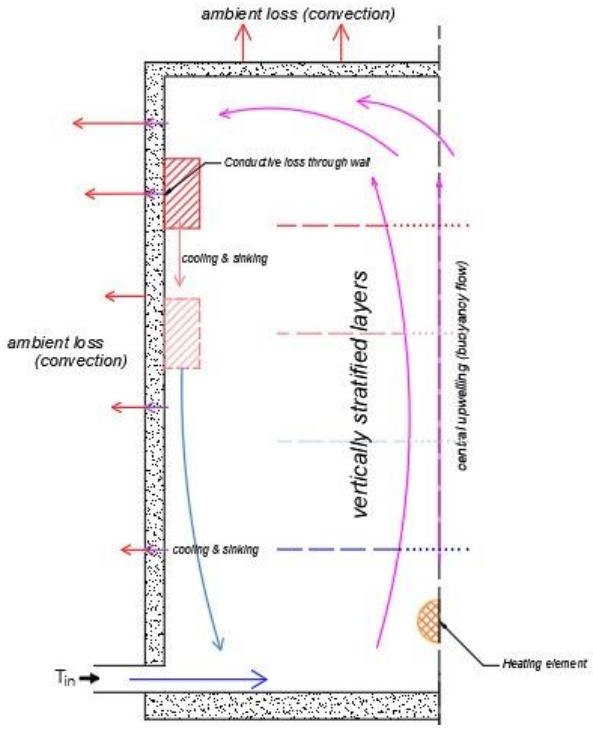


Fig. 35. Illustration of cyclical flows of heat driven by axial conduction inside a hot water cylinder. (Adapted from [110])

Natural stratification is recognised as such an important factor in improving the efficiency of hot water cylinders, that from its recognition in the early 1970s, the phenomenon has been exploited and incorporated into modern designs [119].

It should be noted however that, in comparison to other configurations, directly heated hot water tanks of the kind common in New Zealand households are

considered relatively poor at maintaining stratification due to 'high mixing and turbulence' [117].

Design strategies for enhancing stratification include:

1. Ensuring cold water is introduced mainly at the bottom of the tank and hot water is extracted from the top;
2. Designing tank fittings to ensure cold water enters the tank smoothly to prevent turbulence and mixing with other layers; and
3. Requiring that storage tanks are shaped geometrically to optimise for the conditions conducive to natural stratification, namely being cylindrical and having an appropriate ratio of height to diameter to allow the internal conditions most conducive to stratification [119].

Early experimental work in this field by Lavan and Thompson suggests cylindrical tank height-to-diameter ratio of between 3 and 4 is a good compromise between hot water cylinder efficiency and cost [137].

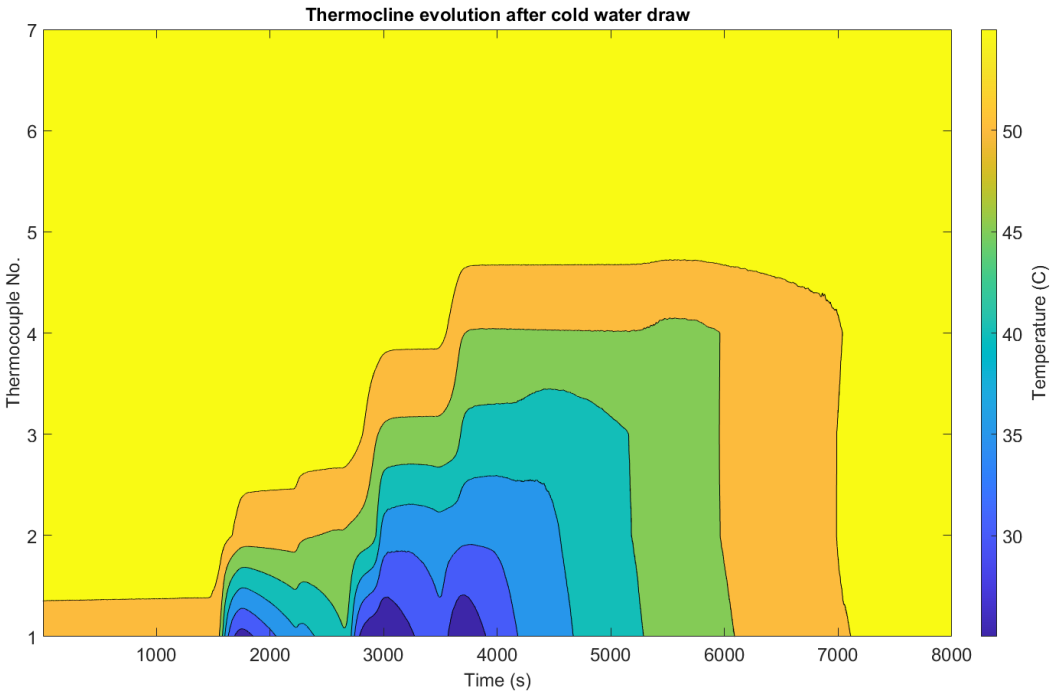


Fig. *36. Measured stratification caused by cold water flow into the base of the Thermann water tank under test in chapter 4. Cold water entering the base of the tank at 1500s, 2800s and 3700s displaces cooler water at the bottom of the tank.*

Buoyancy effects cause the formation of successive thermocline layers, which eventually dissipate as they return to a uniform temperature after time t = 7000s.

Much research has focussed on developing mathematical models capable of predicting the internal states of hot water storage devices, individually and collectively across the grid. These models are based on a deeper understanding of the devices' internal behaviour, especially the formation and evolution of thermoclines and stratified layers.

5.2 Mathematical Modelling of Electric Water Heater Cylinders

Given the internal dynamics of water inside thermal storage devices, various kinds of model have been developed to approximate its behaviour. As is noted by Dumont et al. [116], in many cases the complexity to be modelled can be mathematically intractable, requiring abstraction and simplification of the underlying equations to put them into usable forms.

5.2.1 Description of Common Electric Water Heater Model Categories

Dumont et al. [116] describe eight main categories of mathematical model available for the simulation of thermal energy storage tanks. These range in complexity from the simplest fully-mixed model to intensively complex models using computational fluid dynamics techniques [116].

These are summarised in order of relative computational complexity in

Fig. 37.

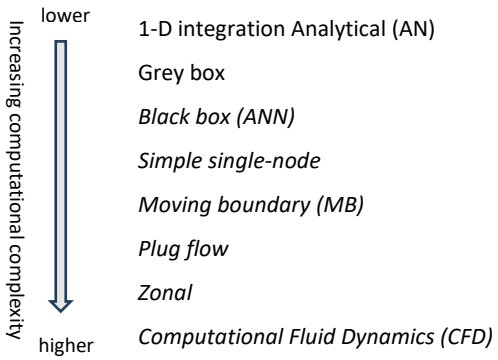


Fig. 37. General classes of EWH model (based on [109]).

The hierarchy of models listed in

Fig. 37 is ranked according to perceived computational complexity and accuracy. This is not however a reflection of utility. In many cases, the most complex models which can simulate the interactions of heat and water in two and three dimensions, may be more accurate, but will be significantly slower due to do the higher computational overheads they require. In contrast, studies required to simulate multiple hot water cylinders to simulate communities and populations may trade off accuracy for speed and tractability by developing simpler models [136].

1. 1-D Integration Analytical (AN)

   One dimensional analytical models are based on the assumption that any temperature variation in the horizontal plane of the hot water cylinder tank is negligible [134]. Zuriga et al. discuss these aspects of one dimensional models in [136]. Since the thermocline is not modelled explicitly in these models, many make use of proxies to account for stratification. The most well-known of these are listed and described in Table 12. The descriptions are based on comments summarised from [116], [117], [119].

Table 12. Common mixing numbers used to characterise stratification in hot water cylinders.

Mixing number	Symbol	Description
Ratio H/D	H/D	Ratio of height to diameter of hot water cylinder tank.
Richardson number	Ri	The ratio of buoyancy forces to mixing forces.
Discharging efficiency ratio	η_d	The ratio of useful heat that can be discharged to the total heat recoverable.
MIX value	MIX	A measure of the energy and vertical temperature distribution in the tank.
Peclet number	Pe	The relationship between bulk heat transfer and conductive heat transfer due to vertical temperature gradients in the tank.
Reynolds number	Re	Comparison between inertial and viscous forces occurring due to the influence of the inlet fluid velocity.

   Of these, the H/D ratio is the simplest metric able to indicate the likelihood that stratification will occur naturally in a given tank [119]. Other researchers have varying opinions as to which of the other metrics is most useful [117], but according to Castell et al., a combination of the Richardson number and

discharging efficiency ratio incorporate all the relevant information needed to take account of the factors leading to natural stratification [119].

Overall, the simplicity and efficiency of the one dimensional analytical model makes them popular when 'long term (standby) storage tank conditions' are to be simulated [117].

2. Grey box model

   This model combines theoretical descriptions of a system based on differential equations, with physically determined parameters that characterise the system based on input and output data [138]. This class of model requires an initial period of system identification to assess important parameters such as tank capacity and thermal mass, which are then incorporated into a simplified state-space based model [139].

3. Black box (ANN) model

   A black box models uses machine learning techniques based on ANNs to approximate the outputs of a system based on measurements of its inputs [116]. A black box model requires large amounts of data to be trained on [138]. Most of the working is concealed in hidden layers, making interpretation of the output opaque and difficult to reconcile to the underlying phenomena as understood based on conventional modelling.

4. Simple single-node model

   The simplest of the multi-node models where the system is described by a single ODE [116]. These models are considered most efficient owing to their low computational overhead [136]. Since the effects of stratification are not included, this comes at the expense of accuracy.

5. Moving Boundary model

   The thermocline is modelled as a Moving Boundary (MB), that divides the tank into two regions on either side of the thermocline [116]. Knowledge of the relative volume of each zone allows for a more accurate calculation of the energy balance in the cylinder, including ambient losses [116]. Mixing

between the two sides of the tank is not considered, as in the model used by Jack et al. [70].

6. Plug flow model

   The plug flow model assumes that cold water enters the base of the hot water cylinder as a single 'isothermal disk'. This occurs at the base of a stack of N similar volumes, all moving upwards towards the outlet [116]. This abstraction of the stratification phenomena in the tank is fast and efficient to calculate, but neglects mixing between the layers. For this reason it is preferred for large tanks with high flow rates and high levels of stratification [116].

7. Multi-node model

   As the name suggests, these models break the volume of the tank into an arbitrary number of layers of uniform horizontal temperature and one-dimensional flow. Increasing the number of nodes increases the computational load. It requires *N* ordinary differential equations to be solved. This allows conductive heat exchange between the stratified layers of the tank to be simulated.

8. Zonal model

   These models represent a compromise between the models described so far and the three-dimensional complexity of modelling the tank using computational fluid dynamics. In this approach, the tank is divided into three dimensional finite volumes to track the mass and energy balance through the volume. As a halfway point between CFD models and multi-node models, these models can capture some physical complexity such as plume entrainment, jets and boundary layer flow [116].

9. Computational Fluid Dynamics (CFD) model

   To fully explore the complexities of water movement inside enclosed storage tanks, finite element methods are used to simulate hot water storage in three dimensions. Detailed modelling of this kind is an expensive investment in time and resources. It may still be cheaper than building working physical models or physical prototypes. Chandra also makes the point that CFD models can be

useful for providing theoretical insights into thermo-fluid behaviour, that can be incorporated into simpler or faster models [117].

This section has described several types of digital EWH model currently used in the field to simulate the behaviour of thermal energy storage tanks. Each has its strengths and weaknesses. Trade-offs between speed and accuracy are needed to choose the most suitable model to use for any given research problem.

Three of these categories will now be examined in more detail, and the equations governing their behaviour described for further comparison.

5.3 Fully-mixed Model

In the trade-off between complexity and accuracy, the fully-mixed model is the most represented in the academic literature due to its simplicity and fast computation. This type of model is especially popular in DSM applications, due to the simplification possible in treating the thermal mass of the tank as a single bulk layer [118]. Chandra et al. also find the one dimensional approach afforded by the fully-mixed model to be useful when long term behaviour or storage tanks needs to be simulated [117].

Examples of their use range from large scale DER management of population levels of hot water cylinders such as in [140], to research on capturing data from individual cylinders for new lines of research using artificial intelligence as in [138].

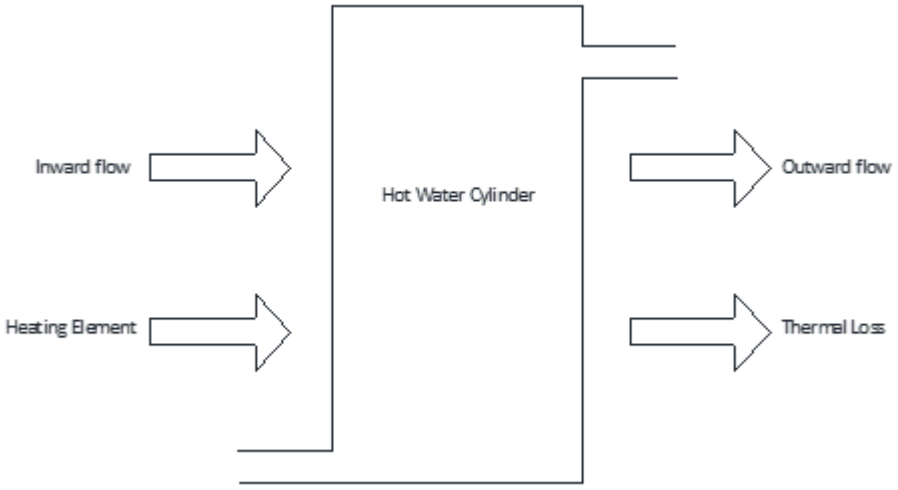


Fig. 38. Simple representation of the energy balance in a residential hot water cylinder.

The simplest representation of this is shown in Fig. 38, where cold water enters the tank at the bottom of the cylinder and is heated by an electric heating element. Hot water is removed from the top of the tank when an occupant of the house turns on a tap to use the stored hot water.

The model assumes heat is uniformly distributed horizontally. On the vertical axis, water entering the bottom of the tank is assumed to mix perfectly with water already present, and the resultant temperature of the water is calculated as the weighted mean to the temperature of the two bodies of water.

Some versions of this approach to modelling, include terms for the convection of heat throughout the water column and the conduction of heat.

In mathematical terms, the physics of the hot water cylinder are modelled as an energy exchange process, with energy external to the tank balanced against the flow of energy into the system. The internal temperature of the hot water cylinder is modelled as a single-node system that can be written as a single differential equation, as shown in the derivation provided by Jack in [70]:

$$C_{\upsilon}\rho V\frac{dT(t)}{dt} = C_{\upsilon}\rho F(t)[T_{in}(t) - T(t)] + Q(t) - UA[T(t) - T_o(t)] \qquad (5.1)$$

Where:

C_{υ} = Heat capacity of water (4180 J/kg·K);
ρ = Density of water (992 kg/m^3);
V= Volume of tank (m^3);

T_o = Atmospheric temperature (°C);
T_{in} = Inlet temperature (°C);
T(t) = Measured water temperature (°C);
F(t) = Volumetric flow rate of water (m^3/s);
UA = overall heat transfer (WK^{-1}). [70]

The rate of change of the temperature of the mass of water in the tank, *T(t),* is equal to the difference between the temperature difference of the mass of new water entering the tank at T_{in}, the energy entering the tank as heat through the heating element *Q(t)* and the heat lost from the tank through the tank walls due to the temperature difference of the tank and the outside air, T_{out}.

This form of the model is called 'fully-mixed' because it neglects the effects of stratification and assumes that the temperature of water throughout the tank to be perfectly uniform. Consequently, it cannot estimate the temperature of water at the outlet as accurately as other models can. Also, this family of model cannot account for the relative height of the thermocline that separates warm water from cold [70]; the point at which a shower might go from warm to cold and when the user will experience most discomfort.

Despite these weaknesses, the fully-mixed model is still seen as good enough for many tasks. It is easy to solve numerically and incurs little in the way of computational overheads to slow computation. Kepplinger for example, claims one version of a simplified mixed layer model as being nearly as accurate as more complex multi-node models, despite the latter being as much as "fifty times more computationally expensive"[118].

5.4 Stratified Model

Moving beyond the simplified single-node model to the two-node model is illustrative of the subtly different approaches to modelling on display in the literature. It is also apparent the effect differing assumptions have on the final form of the equations governing heat storage and transfer and the resulting mathematical models.

Two approaches discussed in this section draw further on the work of Jack et al. in [70], and compare it to the derivation used by Ritchie in [58], [141].

5.4.1 Moving Boundary Example:

To start with, the 'stratified' model used by Jack et al. in [70], based on the single-node mixed model of equation (5.1), is further developed by the addition of a second node to model the upper half of the cylinder. This creates a 'moving boundary' model that splits the hot water cylinder into two distinct zones. These are separated by a 'narrow' thermocline boundary layer [70]. The system is capable of being in one or other of two states: stratified or fully-mixed.

During operation, the flow of cold water into the cylinder (from the bottom of the tank) pushes out an equal volume of hot water from the outlet near the top. At the interface between the new layer of water introduced to the inlet at the bottom of the tank, and the warmer body of water above it, a thin thermocline is assumed to form. The formation of the thermocline splits the tank into two thermally separate parts.

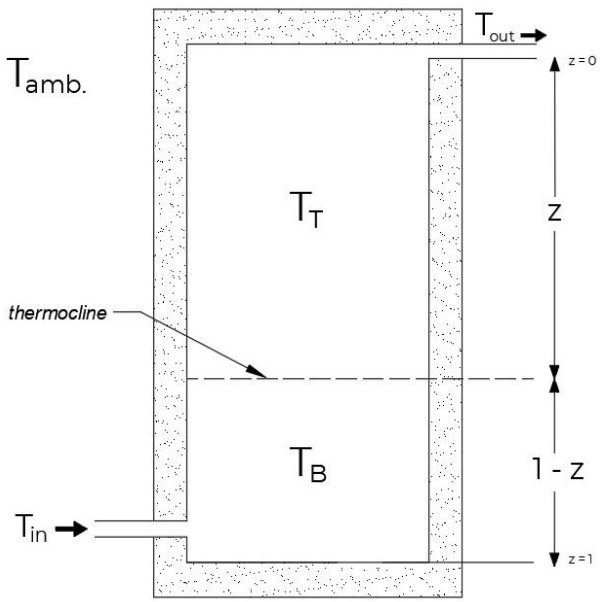


Fig. 39. Schematic of a 2-state nodal model based on a moving boundary (thermocline). Adapted from [63]

When flow ceases, the cooler water in the layer beneath the thermocline is heated by the heating element. In contrast, the top warm layer remains thermally isolated, and slowly loses energy by conduction through the insulated cylinder tank wall to the outside environment.

The mass-balance equation for the lower section of the tank derived in [70] is shown in (5.2). The first term on the right of the equals sign represents the energy associated with new water flowing into the tank. The second term relates to the heat energy provided by the electrical resistance heater. The third term deals with the energy lost through heat loss to the external environment.

$$C_v \rho V_B(t) \frac{dT_B(t)}{dt} = C_v \rho F(t)[T_{in}(t) - T_B(t)] + Q(t) - UA_B(t)[T_B(t) - T_o(t)] \quad (5.2)$$

The form of equation (5.3) is somewhat simpler, because the top section of the cylinder contains no water inlet or source of heat energy. Also, since heat exchange across the thermocline is neglected, the only term in the mass balance equation of (5.3) is related to its ambient heat loss:

$$C_v \rho V_T(t) \frac{dT(t)}{dt} = -UA_T(t)[T_T(t) - T_o(t)] \quad (5.3)$$

The movement of the thermocline height '*z*', is governed by the rate of change of its distance from the top of the tank, which is a function of the volume of cold water flowing in through the inlet at the bottom of the tank, F(t):

$$\frac{dz(t)}{dt} = -\frac{F(t)}{V} \quad (5.4)$$

where the height of the thermocline above the bottom of the tank is defined as:

$$\bar{z}(t) = 1 - z(t) \quad for\ 0 < z < 1 \quad (5.5)$$

The simulation is modelled as a state machine with two distinct states:

1. Fresh flow into the tank (F(i) > 0) causes a new layer to form at the bottom of the tank, separated by a thermocline at distance z from the top of the tank; and

2. The temperature of the lower layer is heated to become hotter than the layer above it.

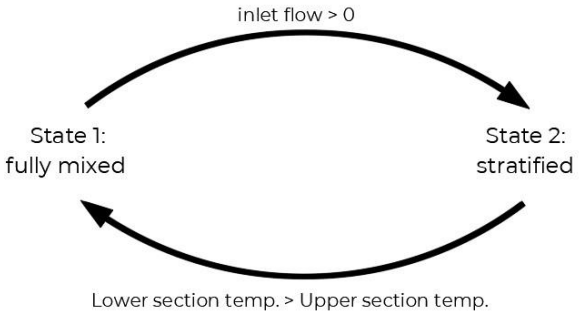


Fig. 40. State-machine diagram of state flow between fully mixed and stratified stated of a 2-state hot water cylinder model.

Only when the bottom layer has heated sufficiently to reach the temperature of the top layer do the two masses of water 'turn-over' and return to the fully mixed state [70]. At this moment, the value of z is reset to zero, indicating the hot water cylinder is fully recharged.

5.4.2 Hybrid-node model

In an alternative derivation by Ritchie et al. [59], the system is treated as a multi-nodal model (N=2), where temperature exchange due to conduction between the top and bottom layers is accounted for.

This derivation is carried out in terms of the total enthalpy contained in the storage tank, in contrast to Jack et al.'s approach which is derived in terms of water temperature.

To represent this, Ritchie et al. use a slightly different derivation, based on a multi-node model but with only two nodes. The subscripts '*L'* and '*U*' denote '*lower*' and '*upper*' respectively [59].

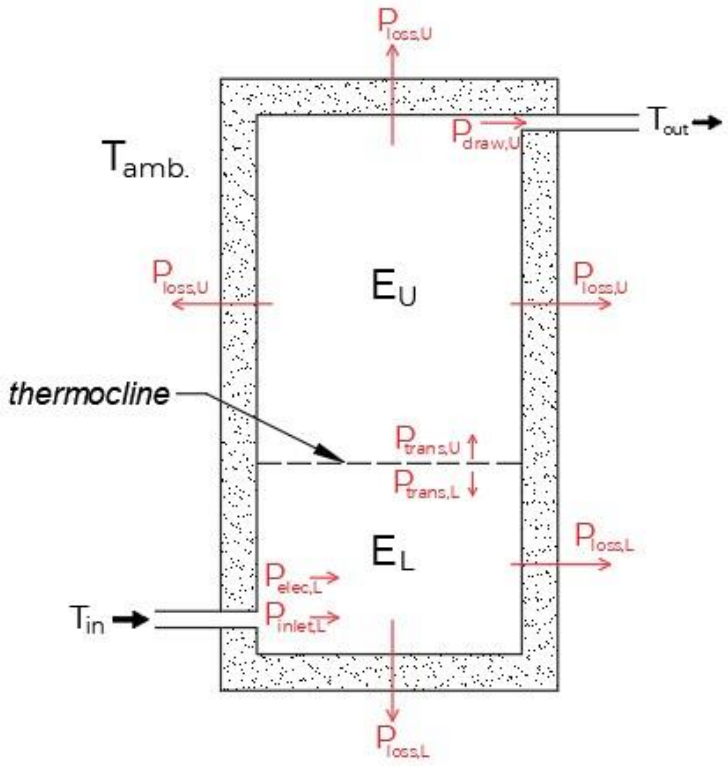


Fig. 41. Schematic of the mass-energy balance of a 2-node hot water cylinder model. (Adapted from [52])

As in the moving boundary model of [70], the equation of the bottom layer includes expressions for incoming water flow and the energy released into the water by the heating element :

$$\dot{E}_{(tank,L)}(t) = P_{elec}(t) - P_{inlet,L}(t) - P_{loss,L}(t) - P_{trans,L}(t) \qquad (5.6)$$

Unlike the expression in (5.2), this derivation includes the component $P_{trans,L}$ to model the flow of energy exchanged across the thermocline, allowing a small amount of mixing to occur. This small modification allows heat to spread vertically through the water column. Models including this variation are described by Kepplinger et al. as 'two-node hybrid' models [118].

In the top section of the tank, the mass energy balance is described by:

$$\dot{E}_{(tank,U)}(t) = -P_{draw,U}(t) - P_{loss,U}(t) - P_{trans,U}(t) \qquad (5.7)$$

In this scheme, rather than explicitly model the height of the thermocline, the energy associated with the bulk storage in the top section of the tank is described by

$P_{draw,U}$, the ambient losses as $P_{loss,U}$, and energy exchanged internally across the thermocline as $P_{trans,U}$.

Whereas the models described by equations (5.1) to (5.5) are concerned with measurements of water temperature, the system defined by equations (5.6) and (5.7) are written in terms of the enthalpy[23] of the system, which is related to temperature by:

$$E_{tank}(t) = c_v \rho V_{tank} T_{tank}(t) \qquad (5.8)$$

The advantages of this approach are twofold. Firstly, by tracking the height of the thermocline, it is possible to track the amount of hot water left in the tank to determine whether the user has run out of hot water and experienced a cold shower.

Secondly, by accounting for two distinct bodies of water, one much hotter than the other, the degree to which heat is lost to the surrounding environment is better estimated. From Newtons' law of cooling, the upper sections of the tank, being hotter, will lose heat at a faster rate. Consequently, they will experience higher standing losses.

The standing losses of the lower part of the tank will be much lower, given the average temperature of this section has already mixed with incoming mains water. This section will already be close to the ambient temperature of the surrounding environment.

An area where there is more uncertainty, is the degree of energy exchange across the thermocline between the two layers.

Ritchie provides a method of estimating the thermal resistance of this layer, and uses an equation that appears to be based on a single-dimensional, lumped element analysis [58]:

[23] Enthalpy (i) is the sum of internal thermal energy (including the latent heat and flow-work) of a substance at constant pressure. It is measured in units J/kg. [122]

$$R_{TH} = \frac{d}{k\,A} \qquad (5.9)$$

Where:

R_{TH} = Thermal resistance of the thermocline;

d = the thickness of the thermocline layer; and

A = the surface area of the boundary layer.

As it is based on a one-dimensional interpretation, the expression must also rely on the conditions across the thermocline to be in a steady state, which is also unlikely to be realistic.

Ritchie does not indicate where the derivation of this expression is sourced. Its use should therefore be considered with some caution. A similar, more rigorous approach to dealing with this problem is provided in [118] by Kepplinger et al. for more detailed multi-node situations, and is the subject of the next section.

5.5 Multi-node Model

The multi-node model is much more computationally taxing than the single-node and two node models mentioned. Detailed examples of this approach are less commonly found the literature, but examples include [141],[142]. These modules are a further extension of the two-node model, except for the fact that heat is permitted to move across each of the thermocline layers.

A simple diagram of the schematic layout of a multi-node model is provided in Fig. 42.

The cylinder is divided amongst N-nodes, which are vertically stacked [141]. To model the most common type of electric hot water cylinders in New Zealand, the bottom node in the stack contains the cold-water inlet and electric heating element.

Energy losses from this segment of the hot water cylinder are modelled as standing losses, separately through the base of the cylinder shell, and the side walls. A further energy exchange is modelled across the thermocline boundary with the tank segment immediately above.

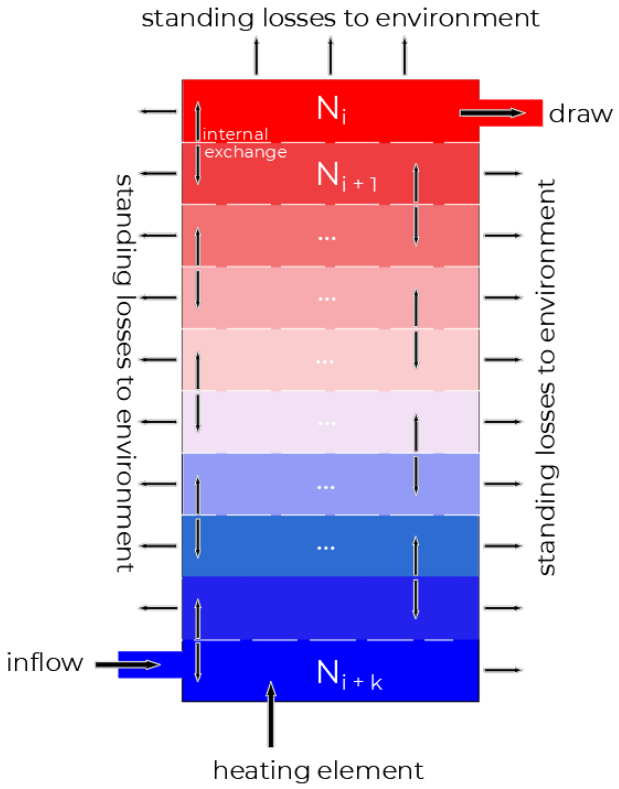


Fig. 42. Schematic of multi-nodal model. The tank is composed of N vertically stacked nodes. Energy flows in/out of each node are labelled with arrows.

Multi-nodal models of this kind become increasingly complicated to simulate, because with each extra node that is added, a new ODE must be added to the system.

Whereas in the simpler systems already discussed have been solved by reducing the systems of ODEs to systems of difference equations, the added complexity of multi-nodal models makes this approach less precise and computationally expensive.

Instead, Kepplinger et al. [118] describes a process of 'homogenisation', in which the system of equations is written in terms of a system matrix **A** that describes the thermal interactions between nodes, a vector of temperatures at each node **T**, and a vector of external inputs, **b**. The reformulation of this system of equations in matrix form becomes [118]:

$$\frac{d\boldsymbol{T}}{dt} = \boldsymbol{AT} + \boldsymbol{b} \qquad (5.10)$$

The 'homogenisation'[24] is achieved by multiplication with $\mathbf{A}^{-1}$, recasting the equation as:

$$\boldsymbol{S} = \boldsymbol{T} + \boldsymbol{A}^{-1}\boldsymbol{b} \tag{5.11}$$

The reformulation allows the $\frac{d\boldsymbol{T}}{dt}$ term to be rewritten in a homogenous form as [118]:

$$\frac{d\boldsymbol{T}}{dt} = \boldsymbol{AS} \tag{5.12}$$

Here, the diagonal elements of matrix A represent temperature values for each node. Off-diagonal values similarly contain terms for the various heat exchanges between layers due to thermal conductivity, as heat is transferred between node layers through the water column

The final analytical solution takes the form [118]:

$$\boldsymbol{T} = e^{\Delta t \boldsymbol{A}}(\boldsymbol{T_o} + \boldsymbol{A}^{-1}\boldsymbol{b}) - \boldsymbol{A}^{-1}\mathbf{b}\ ^{25} \tag{5.13}$$

An implementation of this model is presented in Appendix H. Details of its use in simulation are discussed further in chapter 6. In that section, the model will be compared to solutions generated by numerical analysis techniques.

A feature of this model, and the approach to modelling it using matrices, is that it can be written compactly in code. This compactness makes it easily scaled to any arbitrary number of nodes. As mentioned, the formulation provided by Kepplinger et al. also models the exchange of heat through conduction between layers adjacent to thermoclines [118]. Inclusion of a term for the 'Kronecker delta[26]' ensures that diagonal and non-diagonal elements of the A matrix can be populated efficiently.

[24] A system of equations is 'homogeneous' if the constants on the right hand side of the equations are all equal to zero [143].

[25] In this form, Δt represents the time step used in the discretisation of the continuous system of 5.11. The matrix exponential $e^{\Delta t \boldsymbol{A}}$ advances the system state from time t_o to $t_o + \Delta t$, assuming constant inputs during this interval.

[26] The Kronecker delta function provides a binary output of 0 or 1 depending on the equality of its function inputs, so that δ(m,n) = 0 if m≠n and δ(m,n) = 1 if m=n [144].

Where diagonal elements are present, heat exchange between nodes is then modelled correctly.

A limitation of this approach is that it does not include a mechanism to track the height of the thermocline. If this model were to be developed further in the future, a separate function to track the height of the thermocline could be included, to allow for the temperatures to be reset following heavy water usage events that have depleted the tank.

It is also important to note that the analytical solution in 5.13 is only strictly valid when vector **b** has no time dependence. For practical applications in which temperature is changing with time, the time step chosen must be sufficiently small that **b** can be approximated as constant. The solution is computed using the most recent value of **b**, with initial conditions updated at the next time step. For greater accuracy with rapidly changing inputs, numerical methods like Runge-Kutta may be more appropriate.

5.6 Summary

There are several approaches to modelling residential hot water systems that are available in the scientific literature. A small sample of these have been introduced in this chapter. They range in complexity from the simplest one-dimensional models that assume heat is evenly distributed throughout the cylinder, to more sophisticated three-dimensional approaches.

Following an overview of common categories of hot water modelling approaches, this chapter had focussed on three models based of increasing complexity. It begins with a single-node model based on a single ODE, describing the mass-energy balance of the system. As the simplest of the models introduced, the effects of stratification, including the presence of thermoclines are ignored for the sake of efficient calculation.

In the second example, a stratified, two-node model was explained. This category of model uses a moving thermocline to estimate the height of the boundary between

newly introduced cold water flowing into the tank and the layer of hot water floating above it.

Finally, the system of equations governing the internal dynamics of the hot water cylinder was extended to multiple nodes, allowing the exchange of heat between layers in the tank to be modelled. This provided a more sophisticated approximation of the internal behaviour of the system.

It is anticipated that these examples will form the basis of a toolkit for future research involving the modelling of real hot water systems, which are likely to require some form of model-in-the loop simulation.

Chapter 6 Model Validation

The last chapter summarised common approaches to the mathematical modelling of residential EWHs, including the derivation of single and multi-node models for simple hot water cylinder simulations.

In this chapter, the results of these models will be compared with measured data collected during the experimental results of chapter 4, including a 36-hour water usage simulation involving several water draw events between 5 L and 100 L.

The models' predicted water temperature profiles are compared to actual measurements from a 36-hour test. This comparison determines the minimum model complexity needed to accurately predict the hot water cylinder's state for a given usage profile.

6.1 Optimal Model Size Investigation

Another matter to be explored in this chapter is what level of model complexity is required to accurately represent the temperature profile of working hot water cylinder.

Ritchie et al. [141] have already determined that there are diminishing returns increasing the number of nodes to characterise a hot water cylinder beyond N=4.

Their finding was based on the use of lumped-mass multi-node models, ranging in from N=1 to N=20 nodes [141].

Kepplinger et al. [118] in contrast, base their experimentation on multi-node models set to N=10, which they cite as the value recommended by Duffie and Beckman [145] to provide 'good model accuracy regarding stratification in the tank [118].

Multi-node models were in fact preferred to single-node models by Kepplinger et al., as they were found to have significantly lower estimation error associated with them, at the cost of increased computational complexity [118].

6.2 Effects of Stratification

The main challenge in designing the hot water cylinder models of the last chapter was how best to incorporate the non-linearities seen in the behaviour of real-world hot water cylinders into a one-dimensional mathematical model.

Of the main thermodynamic and fluid dynamics processes occurring inside each hot water cylinder tank, the effects of stratification are the most important to model, as these are most responsible for the efficient nature of hot water storage.

To illustrate the degree of separation possible between differently heated layers within a single hot water tank, Fig. 43 plots measured data from a heating test during the tests described in chapter 4. Starting at an ambient room temperature of approximately 20 ^{o}C, the tank is heated while discrete amounts of water are drawn periodically.

The top graph in Fig. 43 shows temperature measurements taken at different heights on the side wall of the tank. Additionally, the graph includes the ambient air temperature, which is depicted by the horizontal red line.

The lower plot in Fig. 43 shows water draw events of one-minute duration at varying rates of water flow, measured in litres.

The effects of these inflows are seen in the temperature changes measured at the lowest thermocouple in the tank (white line). With each draw, the temperature of the bottom layer plunges to match that of the newly inflowing water mass. Subsequently, the water is heated but remains cooler than the layers above. These continue to heat uniformly until all heating is terminated after approximately 6.5 hours.

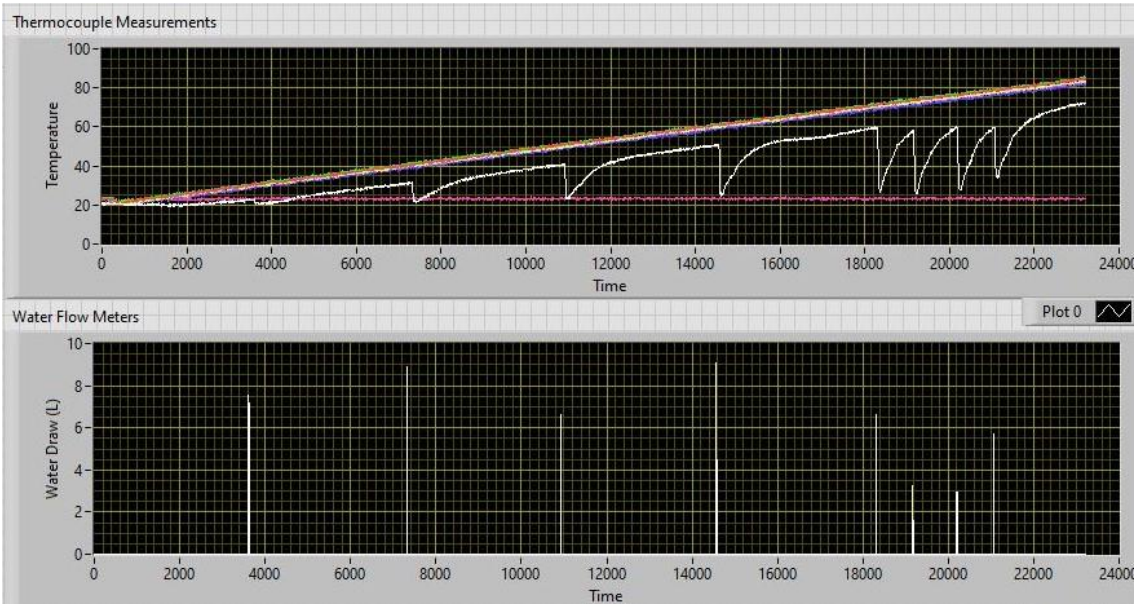


Fig. 43. The stratification of the lower and upper layers in an EWH being recharged at room temperature.

During this process, the trajectory of the heating curve for the upper layers is unaffected by the inflow of cool water to the bottom of the tank. This is evidence of how the upper layers of water in the tank can become thermally isolated from those in the bottom section of the tank, as discussed in the previous chapter.

The implementation of this behaviour in code, with sufficient complexity that it can reproduce the behaviour recorded in the plots of Fig. 43, is the main theme of this section.

Starting with a fully-mixed model, simulations of increasing complexity were compared to a test profile based on a predetermined water draw profile. This was done to understand the trade-off between complexity and accuracy amongst the models and to establish whether complexity leads to improved predictive accuracy.

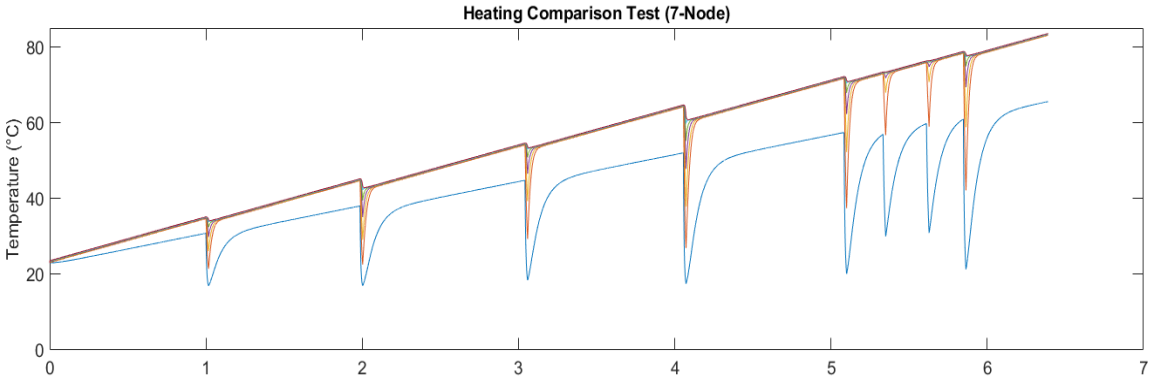


Fig. 44. Example of the output of a computer simulation of the EWH heating experiment shown in Fig. 43, using a multi-node model (N=7).

6.3 Numerical Solutions for Simulation

The practical value of the hot water cylinder models discussed so far can be realised when the ODEs comprising the models are solved and the predictions evaluated. By solving the equations of the system in known environmental conditions, predictions of the system's future states can be made available for analysis.

Solutions to the models introduced so far should theoretically be possible using analytical techniques. For example, the solution of the single node, fully-mixed system is evaluated from equations derived in [70], [118] as:

$$C_{\upsilon}\rho V\frac{dT(t)}{dt} = C_{\upsilon}\rho F(t)[T_{in}(t) - T(t)] + Q(t) - UA[T(t) - T_{o}(t)] \qquad (6.1)$$

Where:

C_{υ} = Heat capacity of water (4180 J/kg·K);
ρ = Density of water (992 kg/m^3);
V= Volume of tank (m^3);
T_o = Atmospheric temperature (°C);
T_{in} = Inlet temperature (°C);
T(t) = Measured water temperature (°C);
F(t) = Volumetric flow rate of water (m^3/s);
UA = overall heat transfer (WK^{-1}).

From this, the analytical solution is an exponentially decreasing family of curves described in [118] by:

$$T(t) = Te^{\left(-\frac{UA}{C_v}(t-t_o)\right)} + \left(1 + e^{\left(-\frac{UA}{C_v}(t-t_o)\right)}\right)\left[\frac{Q}{UA} - \frac{F}{UAV} + T_o(t_o)\right] \quad (6.2)$$

Where t_o represents the beginning time step. [27]

For more complex systems of equations, rather than working through increasingly detailed derivations, a more practical approach is to solve the equation, or system of equations numerically.

Numerical solutions offer the benefit of being relatively straightforward and achievable in contrast to the complexity of an analytical approach. They also provide some surety, given that in many cases, an analytical solution may not be practical to solve [146].

With the exception of the fully-mixed model described above, the other models introduced so far consist of collections of two or more ODEs that operate within 'dynamic' systems [146].

The 'dynamism' occurs because, at any timestep of the simulation, the values of several inputs parameters of the system are subject to change, including those associated with water flow, heating element state and ambient air temperature.

Knowing the starting temperature and other system parameters also allows the system to be solved as an initial value problem (IVP).

Starting with a known initial time and state, the differential equations describing the hot water cylinder are converted into a set of algebraic equations. The solution is then obtained by stepping through these equations from start to finish [146].

At each step, the solution is evaluated and stored for future reference. Unlike the analytical approach however, numerical methods cannot guarantee perfect accuracy. Instead, the solution approximates the actual answer, and care must be taken during

[27] This analytical solution assumes that external temperatures remain constant during the small time interval $[t_o, t]$. For scenarios where external temperatures vary significantly, the solution would need to be computed using sufficiently small time steps or numerical methods.

this process to ensure the choice of step size does not introduce errors detrimental to the accuracy of the final results.

6.3.1 Marching Euler Schemes

The simplest method used to evaluate systems of ODEs numerically is known as the 'Euler' method. In this method, after discretising the system of ODEs by converting them to algebraic difference equations[28] [146], a single fixed step size is chosen to 'march' from the start point to a pre-determined end point. The problem is evaluated at each step by adding all the 'finite differences' encountered. These are used to update the starting point for the next step.

Described as a 'naïve' method by Wilson [146], because of its susceptibility to local errors associated with step size, there are examples of its implementation in the literature. In particular, the method is used by Ritchie [58], whose models also form the basis of the work of others, including Booysen [31] and Cloete [127].

Refinements of the single-step 'Euler' approach exist, and various strategies are available in popular mathematical software packages such as MATLAB.

6.3.2 Ordinary Differential Equation (ODE) Solvers in MATLAB

A practical alternative to the simple Euler method is to modify the step size to make it more responsive to the behaviour of the ODE system under evaluation.

A family of strategies that forms the basis of many of the most prominent solvers available in MATLAB, belongs to the 'Runge-Kutta' family of solutions [146], named after the two German mathematicians who popularised their use in the early 20th century [147].

As with the strategies based on the Euler method, Runge-Kutta based solvers use a single step, but instead make multiple function evaluations at intermediate positions within the step to estimate deviation from the true value [148].

[28] A difference equation is the discrete time analogue of a continuous time differential equation, by approximating differential values $\left(\frac{dy}{dt}\right)$ as elements of finite difference $\left(\frac{\Delta y}{\Delta t}\right)$ [146].

For example, the ‘RK2’ scheme estimates the gradient at the ‘mid-point’ of a pre-determined step and the final value based on that gradient [146].

The more sophisticated ‘RK4’ method, takes this approach further by making four function evaluations inside the step. These are used to establish the ‘average gradient’ across the step, which is used to estimate a more correct final value [146].

A detailed list of solvers is provided by Moler [147], which includes ‘ODE45’, a fourth-order solver that is known as the main workhorse for solvers in the MATLAB package [147].

Apart from the Runge-Kutta solvers (ode23, ODE45), there are also several other algorithms operating on different strategies, many of which are multi-step solvers (ode113), or numerical and backward differentiation formulas (ODE15s) [147].

The solver ODE15s is particularly relevant to the remainder of this section, as the ‘s’ in its name refers to the property of ‘stiffness’ and will be explored in more detail as the various tested models are discussed.

6.3.3 Dealing with System Stiffness

‘Stiffness’ is a concept used in the numerical solution of ODE systems to describe the efficiency with which a solver can find the solution, given the initial conditions and the nature of the inputs available to it.

The term describes the amount of work the solver must carry out to adapt to solutions that may change slowly, against a background of more rapid changes occurring locally.

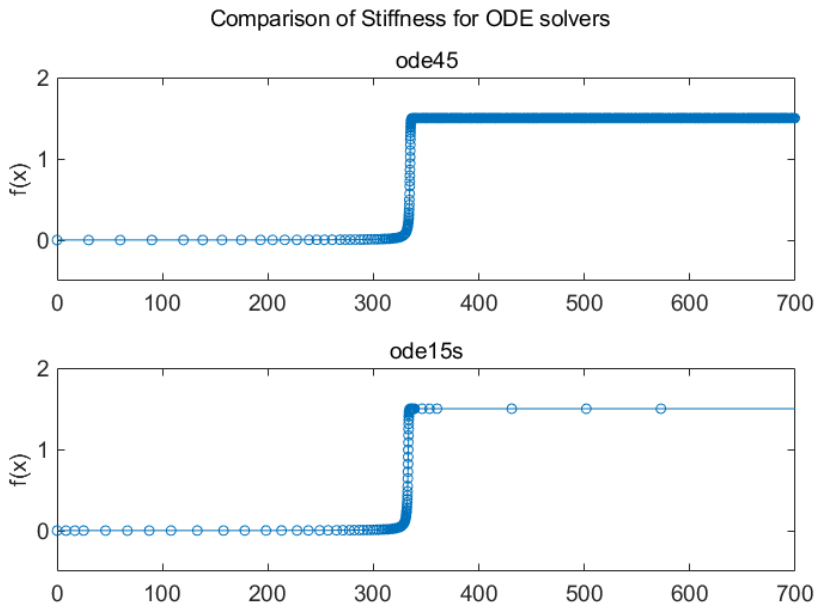


Fig. 45. A comparison of ODE solver reaction to system stiffness associated with the step function $f(x) = 3x^2 - 2x^3$. *In the upper plot, the ODE45 solution does not adapt to the new steady state, whereas the ODE15s solver is not similarly affected. (Adapted from [147])*

Encountering the square edge of a step function, such as depicted at the turning points of the step function in Fig. 45, the solver is forced to drastically reduce its step size at the point the function steps from one state to the next. The ODE45 solver in the upper plot reduces its step size to match the local change but is too 'stiff' to adapt to the steady state encountered after 350 seconds [147]. The small step size remains, and the solver performs many more calculations than necessary, increasing the solution time. The specialised ODE15s solver in comparison, can adapt its step size to the new steady state and solve the system of equations efficiently.

This example applies to the hot water cylinder models and testing described in Chapters 4 and 5. This is because the data relating to the water heating element and water flow meters are in binary states of 'fully on' or 'fully off'. The square shaped step profiles can cause the ODE solvers to behave in the same 'stiff' manner as the example described in the upper section of Fig. 45.

Overall, the effects of system stiffness are also demonstrated by comparing the total solution times of several MATLAB ode solvers tasked with solving the system of equations associated with a sample daily profile for a three-node hot water model of the type discussed in chapter 5.4.2.

In this test, the ODE solvers listed in Fig. 46 were each tasked with solving the ODE model for daily consumption using maximum time steps of 1 second, 10 seconds, 60 seconds. In the final test, there was no maximum step size.

The times taken to solve the system of equations were recorded and plotted on the y-axis of Fig. 46. In all cases, the 'stiff' solvers ode23s, ode23tb and ODE15s performed best, indicating that the system is indeed a 'stiff' system [146].

As the solver most suited to solving for stiff systems, ODE15s was consistently the fastest to carry out the daily simulations. For this reason, it was chosen as the main solver for the remainder of the simulations and tests carried out in this chapter.

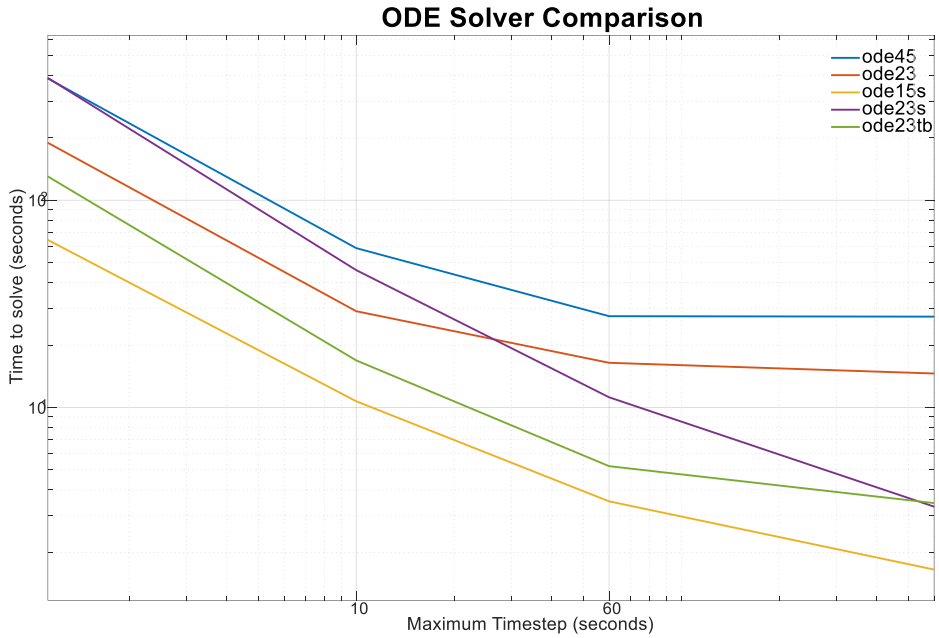


Fig. 46. Comparison of the time taken to solve the N=3 system of ODEs for various MATLAB ode solvers. The solvers 'ODE15s' and 'ode23s' performed fastest for all allowed maximum timesteps.

6.4 Comparison of Solving Schemes with Test Data

Having described several types of hot water cylinder models of varying degrees of complexity in chapter 5, this section aims to determine how accurately each is able to predict the thermal profile of the experimental hot water cylinder following an arbitrary daily hot water consumption pattern.

The models tested include:

- the simple single-node (fully mixed) model;
- the two-node (stratified) model;
- three-node (N=3) model;
- seven-node (N=7) model; and
- ten-node (N=10) model

The hot water consumption profile adopted is based on a flow profile measured in the experimental research phase over a 36-hour period, which included an overnight stretch during which no water was withdrawn from the tank.

The load profile is plotted in Fig. 47 and shows the measured thermocouple temperatures recorded during the 36-hour testing period. The test period lasted from 22:00 hrs on 12/11/23 until 10:00 hrs on 14/11/23.

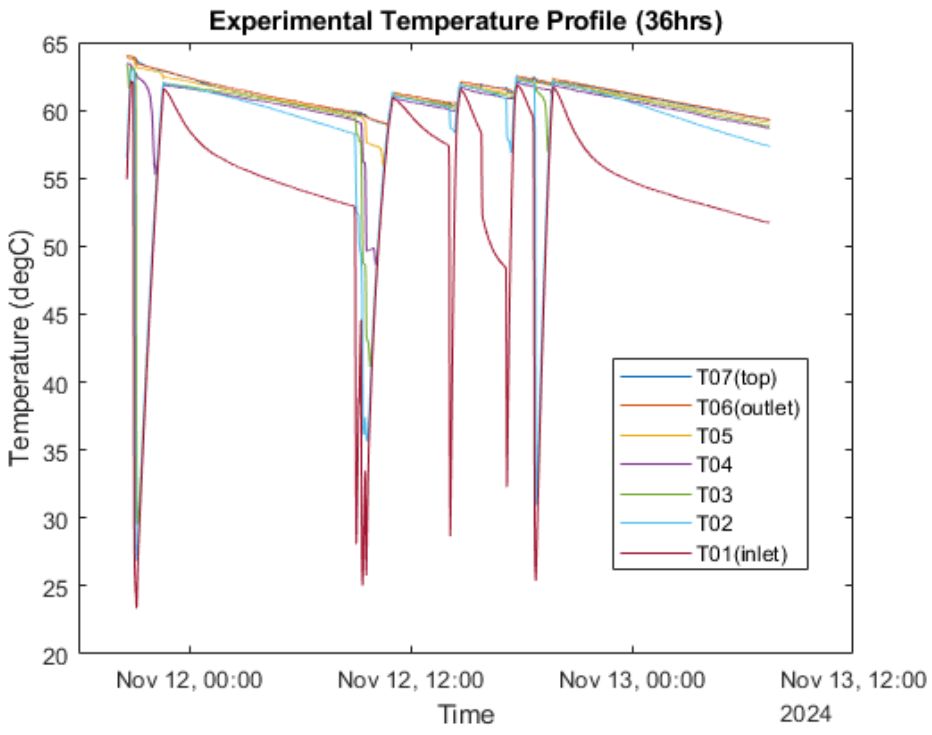


Fig. 47. The results of an experimental water consumption profile carried out over 36 hours.

The times and quantities of water removal from the tank are also recorded in Table 13 . A standard flow rate of 9 L/min was used in all water draw events:

Table 13. Water draw event schedule for hot water cylinder model test.

Draw No.	Start time (hh:mm)	End time (hh:mm)	Duration (sec)	Total volume (L)
1	20:38	20:49	680	**100**
2	09:00	09:06	340	**50**
3	09:13	09:14	34	**5**
4	09:21	09:27	340	**50**
5	09:34	09:39	204	**30**
6	14:06	14:09	204	**30**
7	15:52	15:53	68	**10**
8	17:12	17:14	136	**20**
9	18:41	18:49	476	**70**

For the models listed, a comparison of the accuracy of the estimated temperature of the top layer at the hot water cylinder outlet was made with the measured temperature of the experimental hot water cylinder.

The root mean square temperature difference (RMSE), similar to the approach taken by Kepplinger et al. [118] to determine how well a model matched experimental data.

The RMSE value is compared for simulations based on the simplest Euler marching scheme against the solver ODE15s.

Usage tests profiles showing measured data on which the comparison is based are provided for models with 1, 2, 3, 7 and 10 nodes. Larger images showing the full simulation and code for the model tested are provided in Appendix E – Appendix H.

6.4.1 Single-node Model (N=1)

The results of two separate simulations are shown here for the single-node model, solved using an Euler marching scheme and MATLAB's ODE15s solver

Table 14. Single-node simulation metrics.

Simulation type	RMSE	Time (s)
Euler	8.3	0.04
ODE15s	3.3	0.6

The profile tracked by the ODE15s solution is significantly closer to the outlet temperature of the experimentally measured tank, as shown in a comparison of the profiles plotted in Fig. 48 and Fig. 49.

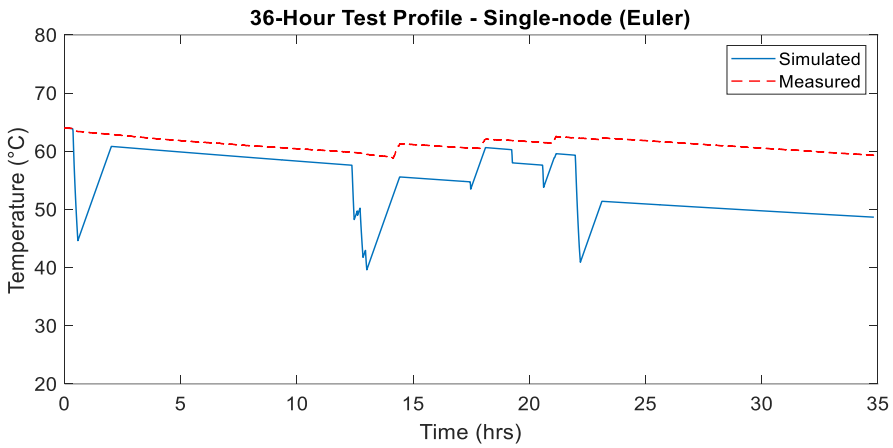


Fig. 48. The estimated and measured temperature of the hot water cylinder after a 36-hours simulation using the demand profile in 6.4.

The profile generated by the Euler solution in contrast more closely tracks the specific water draw events of the input data, possibly due to the fixed step size of the Euler solution.

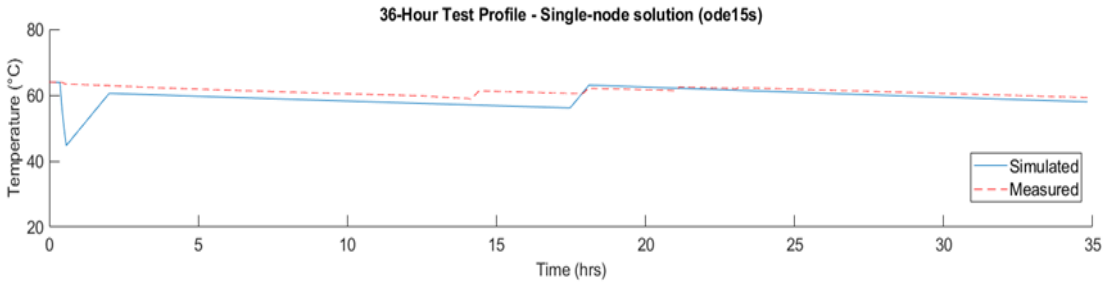


Fig. 49. The ODE15s solution of the same simulation setup. The ODE15s solution tracks the measured temperature more closely than the Euler solution, but not the water draw profile.

The initial heavy draw of 100 L from the tank is also a problematic feature for both models. On both Euler and ODE15s simulations, the RMSE spikes to approximately eighteen, with an absolute temperature of approximately 45 degrees Celsius, about half the value measure in the response of the test cylinder.

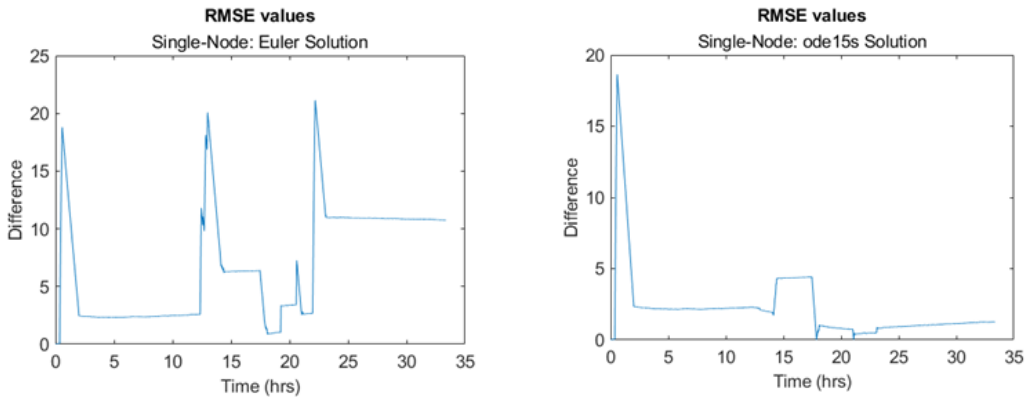


Fig. 50. RMSE plots for Euler (left) and ODE15s (right) solutions to the single-node simulation for 36-hours.

In comparison to each other, there is a significant difference in speed, suggesting the stiffness of the problem remains a factor to be considered, even with the selection of a specialised stiff solver like ODE15s.

However, despite being fifteen times faster than the ODE15s solution, the overall Euler solution drifts with time. The final water temperature estimate is more than 12 degrees Celsius below the actual temperature measured in the tank at the end of the testing period.

Still useful as a rough proxy for average water tank temperature, on the basis of the testing that was carried out, the single-node, fully-mixed hot water cylinder model omits too many characteristics of hot water heating behaviour inside the tank to be practically useful as reliable model.

6.4.2 Mixed-node Stratified Model (N=2)

The performance of the two-node, stratified hot water cylinder model is described next, based on the results of its estimation of the same hot water draw profile.

This model differs from the single-node model tested in 6.4.1, as it uses three ODEs to model the system in one of two states. In the initial state, the cylinder is taken to be fully mixed, and the simulation runs using the same model as demonstrated in Section 5.3.

The model transitions to the second state at the introduction of cold-water flow into the bottom node. The system is represented by two ODEs, for the lower and upper sections of the tank respectively.

The temperatures of the upper and lower sections of the tank are plotted in red and blue respectively in Fig. 51 and Fig. 52.

The improved representation of the two-node model over the single-node model of the previous section is clearly visible, and simulations using Euler and ODE15s schemes both measure improved RMSE values when compared to earlier experimental measurements for the profile described in Table 13.

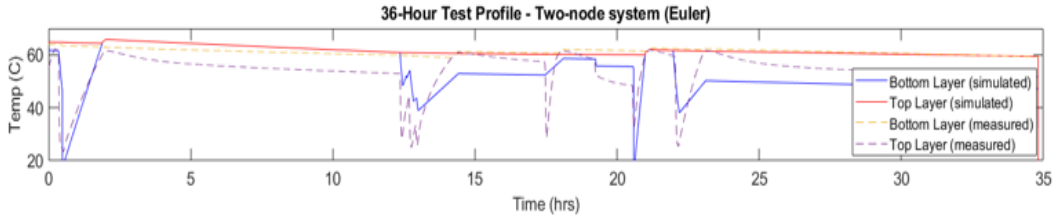


Fig. 51. Comparison of the two-node Euler solution with the measured outlet temperature.

A comparison of solution algorithms, between the Euler and ODE15s methods shows similar results to those seen in the single-node example, though this isn't immediately evident in the measured RMSE values shown in Table 15. Whereas the ODE15s simulation better represents the convergence in temperature between top and bottom layers, albeit with a slightly higher RMSE value.

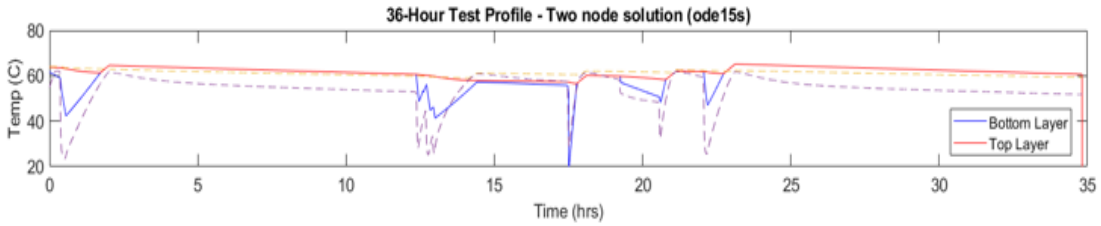


Fig. 52. Comparison of two-node ode15s solution with the measured outlet temperature.

The development of the temperature profile of the bottom node in Fig. 51 resembles the temperature profile seen in the single-node model of Fig. 48. Consequently, the simulated temperature of the bottom layer reaches the temperature of the top layer on heating on only two occasions, whereas the ODE15s simulation better represents

the convergence in temperature between top and bottom layers, albeit with a slightly higher RMSE value.

Table 15. Two-node simulation metrics.

Simulation type	RMSE	Time (s)
Euler	1.4	0.71
ODE15s	1.99	58

The effect of adding a second node however, increases the complexity of the code for the model. The added complexity of introducing terms to track the thermocline height, the state machine to transition between fully-mixed and stratified states, as well as the extra ODEs required, have pushed the elapsed time for the Euler solution to 58 seconds. The elapsed simulation time for the ODE15s solution was 82 times slower, perhaps also caused by system stiffness and its effects on the variable step solution of the model. The code for this model is presented in Appendix F.

6.4.3 Muti-node Model (N=3)

Following on from the moving boundary approach of the two-node model, the addition of third layer to create three equally sized volumes in the tank proved challenging and the results were not satisfactory.

The extra complexity caused by the extra layer developed in part because, with each additional node, an additional state and its associated transition logic were added to the model. Developing the moving boundary further using the same architecture quickly became untenable, due to the need to track the height of the thermocline through multiple states and the extra logic (consisting of if…else) statements required to do this.

In comparison to the code of the two-node model (Appendix F), the number of lines of code required to model the three-node moving boundary model nearly doubled, increasing from approximately 230 lines to around 380 lines of MATLAB code.

Neither did the three-node model improve significantly on the performance of the two-node model. Partly, this is because the addition of an extra node was arbitrary and didn’t model any significant features of hot water cylinder physics. Contrast this

with the benefits of adding the second node to the full-mixed model, which allowed modelling of the thermocline, whose position enabled rudimentary simulation of the position of the main determinant of water tank performance, namely the height of the thermocline.

Instead, adding a third, fourth and fifth nodes to the model generated increasing complexity due to the growing number of states and differential equations.

The code developed for this model is presented in Appendix G.

Fig. 53. Example of the output generated by a three-node moving boundary model and an Euler marching scheme. The output of this model was disappointing and did not improve on the simulated output of the two-node stratified model.

An example of the output of a three-node model is shown in Fig. 53, which demonstrates many of the features seen in the two-node model. Flaws in the design and implementation of the code however reduced the usefulness of this model design and rendered it unworthy of further development.

6.4.4 Multi-node model (N=7)

Owing to the difficulties encountered when extending the complexity of moving boundary models beyond N > 3, particularly the increase in code length and number of function calls, the structure of the simulations models was changed to be more like the analytic model demonstrated by Kepplinger in [118].

This approach was described in chapter 5.5 and uses linear algebra techniques to formulate the behaviour of each node in the system as an extra row in a matrix diagonal matrix corresponding to node temperatures.

With models solved numerically using Euler's method or MATLAB's native ODE solvers, the N layers of the hot water cylinder are represented as N separate ODE's that are looped through at each timestep to calculate the outlet water temperature, T.

In this example, a seven-node model was chosen to match the seven sensors installed in the laboratory phase of testing. The model was simulated three times, using all three methods of equation solving.

The results for this test are shown in Table 16, where the RMSE value and time duration of the simulation in seconds are recorded.

Table 16. Seven node simulation test result metrics.

Simulation type	**RMSE**	**Time (s)**
Analytic	2.7	3.8
Euler	8.3	0.04
ODE15s	2.3	0.6

As with the simpler two-node model of section 6.4.2, the solution using Euler's method was found to be significantly faster than the time required to compute solutions using the analytical and variable step solvers, even with the timestep set at 1 second.

However, this also resulted in the least accurate solution, which was expected given the likelihood of minor errors to compound with each step or at longer times steps. In comparison, the analytic solution plotted in Fig. 54 took nearly four seconds to converge but had better agreement with the measured data.

Similarly, the ODE15s solution was slightly slower than the Euler solution but made up for the delay with extra accuracy. This was expected given the solver's ability to modify its timestep to ensure the best balance between speed and accuracy.

The code used to solve a seven-node model using this analytic matrix approach is presented in Appendix H alongside the code for Euler and ODE15s solutions.

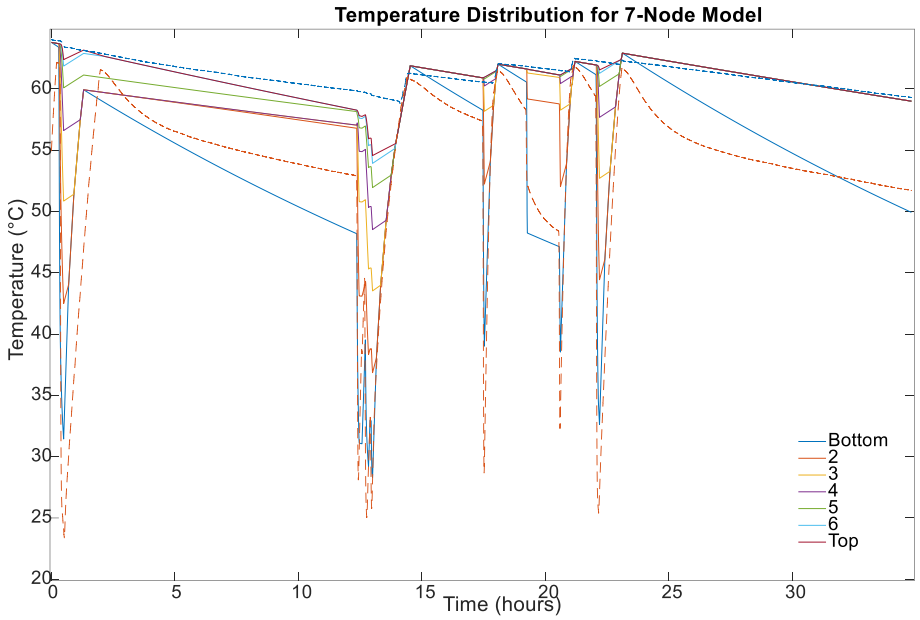


Fig. 54. Analytic solution of a 7-node model compared to measured values of temperature for the 36-hour test (shown as dotted lines in the plot).

6.4.5 Multi-node Model (N=10)

The same test is carried out here for a ten-node version of the same model, to gauge the effect of increased complexity on the duration of the solver time.

Table 17. Ten-node simulation result metrics.

Simulation type	RMSE	Time (s)
Analytic	5.3	4.5
Euler	8.3	0.04
ODE15s	1.7	67

As with the previous section, the RMSE is also calculated, based on a comparison of the measured data of the single day simulated household profile carried out during the laboratory test phase.

The results in this case shared the same basic pattern as the seven-node scenario. The Euler solution was an order of magnitude faster than the analytic case, which was in turn faster than the solution of the ODE15s solver (Fig. 56), which may have been affected by system stiffness.

The ODE solver solution was the closest to the actual measured temperature profile, which in this case had an RMSE value of only 1.7.

Since these simulations were carried out only once for each model, they cannot be considered as indicative of the actual accuracy of the model as it has been coded.

A more proper validation would require many repeated tests of different flow profiles across many days to prove the accuracy of these models and that they were correctly predicting the performance of the hot water cylinder under test.

The N-node model demonstrated here is however, a more detailed attempt at simulating the internal dynamics of the hot water cylinder than the fully mixed and stratified models.

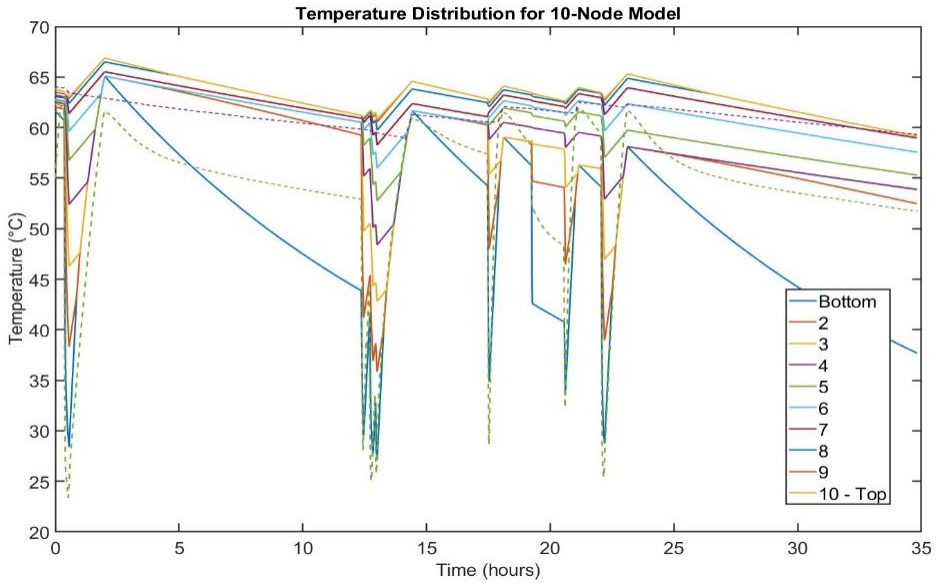


Fig. 55. Comparison of the temperature distribution of a 10-node model using the analytical approach described by Kepplinger et al. with the temperature values measured during the 36-hr laboratory test (dotted lines).

The main difference is that heat flow between layers inside the tank is modelled, by allowing heat to flow in the direction of the thermal gradient. This is achieved through the inclusion of a Kronecker Delta function in the code, as well as variable to estimate the thermal conduction between layers inside the tank.

On the basis of this single test, it is not possible to properly assess whether the move to more layers, such as from seven to ten as in this example offers a performance

improvement in proportion to the extra computation required. It does suggest however, that there are diminishing returns in extending the model further than this.

Adding nodes reduces the thickness of the layers, but increases the uncertainty and error associated with the heat exchanged between thermoclines, which might be compounded further by computational errors associated with the choice of solver. Compounding these sources of error could negate any benefits of increasing the model's complexity. Neither would adding more layers address some of the less dominant effects touched on in chapter 5.

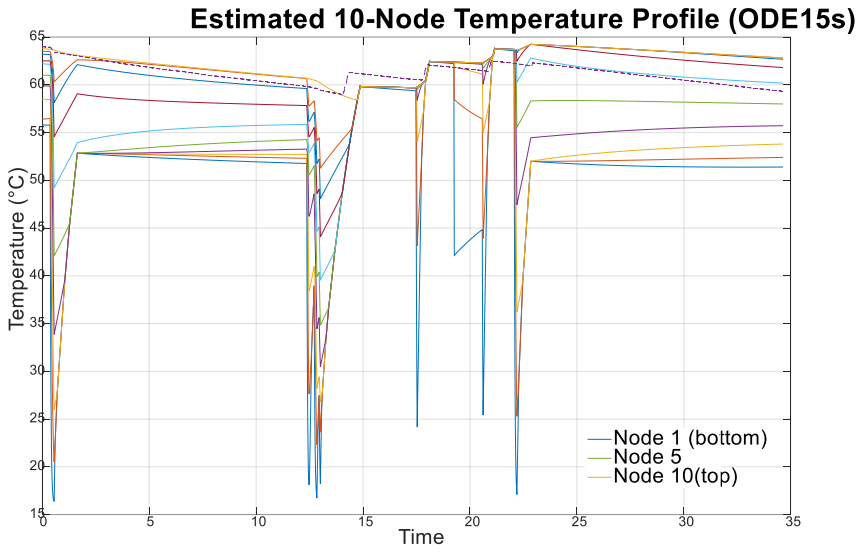


Fig. 56. Comparison of the temperature distribution of a 10-node model using ODE15s, with temperature values measured during the 36-hr laboratory test (dotted lines).

6.5 Summary

This chapter examined three primary methods of solving systems of ordinary differential equations (ODEs) to describe the behaviour of several hot water cylinder systems. The chapter focus mainly on marching Euler schemes and ODE solvers, but two additional examples of analytical solutions for first and multi-node systems were also discussed.

An important step was made in identifying hot water cylinder systems as 'stiff' systems, on account of the sudden changes in water temperature associated with the on/off control of the hot water heating element.

An explanation of ‘stiffness’ in systems of ODEs was provided in the context of hot water control, and this feature of the system was used to select the solver ODE15s from amongst the many solver options available in the MATLAB computing environment.

Solutions to the three solution types were plotted and compared graphically with the measured outlet temperature data recorded over several days, beginning with the simplest single node model.

Overall, solutions employing the simple Euler marching scheme were found to execute fastest, but with the least accuracy. In contrast, solutions employing the ODE15s solver were slightly slower but more closely aligned to the measured outputs of the test profile being simulated.

The analytical model tested proved promising and was able to solve for the seven-node and ten-node systems with similar accuracy to the ODE15s solver. However, being one-off tests of a single water use profile, the limitations of this testing procedure were acknowledged. More rigorous testing under a range of different scenarios and user profiles should be carried out in the future to gather a more detailed understanding of the reliability of the various models tested and their ability to accurately predict the state of the hot water cylinder system over many hours and days.

Finally, as an overall assessment of the relative merits of simple and complex hot water cylinder models, no real advantages were seen in choosing complex systems over shorter more simple ones.

The solutions times for larger models, particularly the ten-node model were of significantly longer duration than other options, but without offering any obvious advantages as far as the profile tested here was concerned.

The two-node model on the other hand, had the advantages of speed, simplicity and ease of implementation. It offered the speed and simplicity of single node model, but with a rudimentary representation of thermocline height tracking as a proxy for stratification. Since the height of the thermocline is tracked in this model, it could

also be easily modified to estimate the state of charge (SOC) of the hot water cylinder.

Its simplicity could also facilitate its incorporation into a simple mode-based control system like the one I explored in chapter seven, or a more comprehensive control method like reinforcement learning.

Chapter 7 Optimal Control Approaches

Moving beyond the simplest forms of hot water control, whether simple thermostat, timed or manually controlled, requires a new approach that incorporates some degree of intelligence into the hot water storage device.

Many of the proposed hot-water control approaches in the literature rely on intensive computational methods, often involving external solvers and complicated AI algorithms, which are unlikely to be easily implemented on a simple hardware platform.

This section describes three alternative solutions for automating the decision-making process for when a hot water cylinder should activate and draw power from the electricity grid to heat water.

This will be achieved by turning the physical question of heating water into a mathematical problem of 'search' in an unknown state space.

Using this approach, the problem of searching for the shortest pathway through a state-space then becomes analogous to searching for the optimal heating schedule to be undertaken. In these spaces, the parameters of the system that are to be included for optimisation, such as water temperature, heating cost and user comfort.

This chapter begins with a brief description of Dynamic Programming (DP), which forms the basis of one possible solution to finding simple, but partially intelligent solutions to automated hot water control.

7.1 Dynamic Programming (DP) as a Guide to Hot Water Control

Numerous technologies have been used to address the question of when the decision to heat water should be made. These vary from simple scheduled approaches to complex, pattern-finding approaches employing methods developed in the advanced control and machine learning communities.

To avoid becoming bogged down in the complexity and costs associated with advanced control techniques such as model predictive control (MPC), deep

reinforcement learning and other equally complex algorithms, the main focus of this work has been how scheduling might occur in a way optimal to the specific environment experienced by a hot water cylinder and user at a specific time, subject to how that environment might change through the day.

The term 'Dynamic Programming' was originally coined by Richard Bellman in the early 1950s to describe finding optimal solutions to combinatorial decision-making problems [149].

An important element of Bellman's strategy is to take large, complicated problems and break them into smaller, more easily solved 'single-stage' sub-problems [150].

Then, working backwards from a terminal state through every single state 'tail sub-problem', the *principle of optimality*[29] is invoked to decide the best step to make through the search space [151]. The search is terminated when the initial state is reached, and each decision taken during the search is collated and re-cast as the control step of the optimal decision sequence [151].

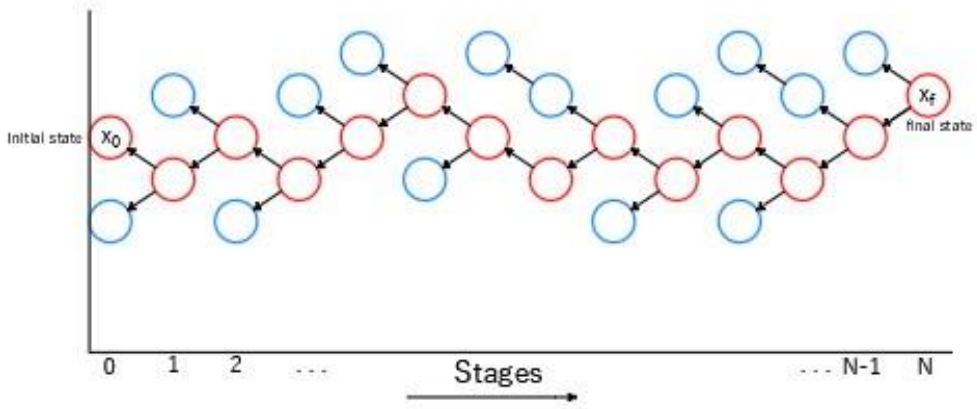


Fig. 57. Dynamic programming backwards search. The red circles shows how optimally connected nodes ('tail sub-problems') connect to find the optimal pathway in the search space, starting from the initial state x_0, and working back to the final state, x_f (adapted from [151]).

[29] The principle of optimality as described by Bertsekas, states that each of the sub-steps that combine to form an optimal path must also themselves be optimal. Accordingly, the "tail of an optimal sequence is optimal for the tail subproblem". [151]

An illustration of the process is depicted in Fig. 57. Starting the search at final state x_f, the DP algorithm searches through a tree of possible predecessor states (indicated by blue circles) along a path of lowest cost, to reach the initial state x_0.

The path from each predecessor state to the next is a 'tail subproblem', which must be chosen optimally if the optimal path through the sequence is to be found.

7.2 Application to Optimal Hot Water Scheduling

Electric hot water cylinders in New Zealand are predominantly of the resistive heating element type, employing a thermostat to determine when water should be heated [20]. Consequently, the heating element is always in one of two states: fully on or entirely off. The binary state of the heating element makes the control of cylinders a combinatorial optimisation problem potentially suited to a control approach employing DP [58].

7.3 Implementation in MATLAB

Adapting the problem of hot water control to this control approach requires discretising the continuous fully mixed hot water cylinder model described in chapter 5.3, following the notation offered in [25][30]:

$$T_{j,n+1} = T_{i,n} - \frac{1}{(C\upsilon\rho V)}[f_n(T_{inlet} - T_{i,n}) + q_n + \frac{UA}{(C\upsilon\rho V)}(T_o - T_{i,n})]\Delta t \qquad (7.1)$$

Where:

C_υ = heat capacity of water (J/kg·K);
ρ = density of water (kg/m^3);
V = volume of tank (m^3);
$q_{i,n}$ = electric heating (W);
f_n = volumetric flow rate of water at step n (m^3/s);
UA = heat transfer through the tank (WK^{-1});
n = time step index, n ϵ {1,2,…,K};
T_{inlet} = Inlet temperature (*°C)*;
T_o = Ambient air temperature (*°C)*;
$T_{i,n}$ = ith tank water temperature at time step n+1 (*°C)*;
$T_{j,n+1}$ = jth water temperature at time step n+1 (*°C);*
Δt = time step.duration

[30] In this approach, the system is modelled as a graph where nodes represent discrete possible temperature states ($T_{i,n}$) of the EWH at each time step n. Edges connect adjacent nodes $T_{i,n}$ and $T_{j,n+1}$, and represent heating and cooling between each step. Each edge connects the two nodes, and describes the physically realisable temperature transitions between adjacent time steps (n, n+1), according to equation 7.1.

Calculating the temperature of water at the next time step, for each of the possible binary states of the heating element ('on' or 'off'), generates a sequence of branching steps that can be visualised in a similar way to the path illustrated in Fig. 56.

The branching nature of DP approaches causes an exponential growth factor of $O(2^N)$ in the number of calculations at each time-step[31], rapidly inflating the search tree to fill up computer memory if not handled carefully.

To demonstrate the effectiveness of DP when applied in this context, the simplified fully-mixed model defined in equation 7.1 was coded in MATLAB and is provided for reference in Appendix J.

The 'curse of dimensionality' is a term which refers to the rapid increase in the degree and complexity of a DP solution as the number of permutations to be calculated grows with successive time-steps.

The term is often used in relation to problems of this kind, owing to the exponential nature of the branching factor [59]. For this reason, no attempt was made to simulate the two-node 'stratified' model described in chapter 5.4 using this technique, because of the unrealisable computational cost this approach would cause.

Results for a typical solution to the optimal scheduling problem solved by the algorithm can be seen Fig. 58. Using data collected from House 31 of the GREEN Grid project, the algorithm has taken as its main input expected hot water flow data (row 4) for a day in June and calculated the most appropriate times the heating element should be turned on (row 3). The hot water temperature shown in row 1 stays above the minimum comfort threshold of 40 ^{o}C throughout the day, ensuring a level of comfort to the user is enforced. A noticeable ramp-up in temperature occurs at around step 40 (approx. 4 pm) to preheat the cylinder for a large water draw at step 42.

[31] "big-Oh" notation describes the estimated upper bound on the running time of an algorithm [152]. For example, an O(N) algorithm describes linear growth in the running time of a program, whereas an $O(N^2)$ describes a system using exponentially more resources with each time-step.

This has the added benefit of heating the water past the 60 ^{o}C threshold required to protect against the growth of legionnaire's disease [153].

The algorithm also pays attention to the dynamic price of electricity at the time and date concerned in deciding whether to turn it on at any timestep. In this case, the wholesale spot price for electricity is used, but this is more likely to be a price set by the local supplier.

Currently, the algorithm takes approximately two minutes to develop a schedule of 24 hours of use, at a timestep resolution of 900 seconds (15 minutes).

An attempt to improve on this performance was made next using Dijkstra's algorithm.

7.4 Shortest Path Graph Search: Dijkstra's Algorithm

Dijkstra's algorithm is named after the Dutch computer scientist Edsger Dijkstra in 1959 [154]. Dijkstra's algorithm is also a form of DP, but unlike the brute force approach of the last section, Dijkstra's algorithm is able to exploit key characteristics of directed graphs and priority queues to systematically search a space and compare the relative value of the options available to selectively focussing its search on the most promising regions of the search space [53].

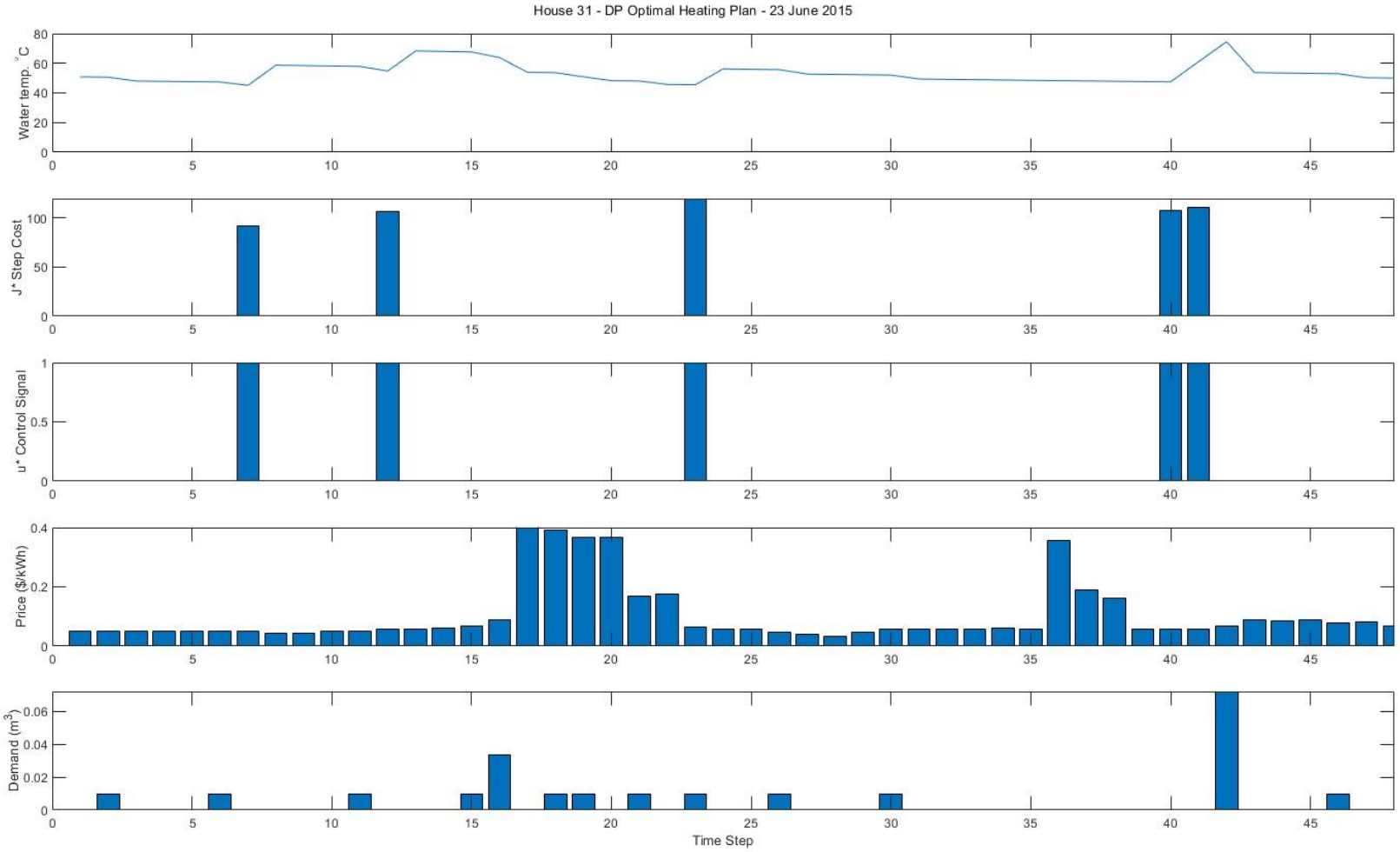


Fig. 58. Heating schedule for a single day using DP algorithm. (House 31 on 23 June 2015).

7.5 Weighted Acyclic Graphs

Dijkstra's algorithm is often associated with searches in 2-D grid spaces, such as the grid spaces and mazes in computer games, or as the basis for navigational search. As a method for discovering optimal heating profiles for hot water, it has been adapted by Kapsalis and Hadellis [25]. In this approach, the discretisation of the continuous hot water cylinder model also allows the search space to be discretised into an explicit search space of connected nodes on a directed acyclic graph (DAG). [32]

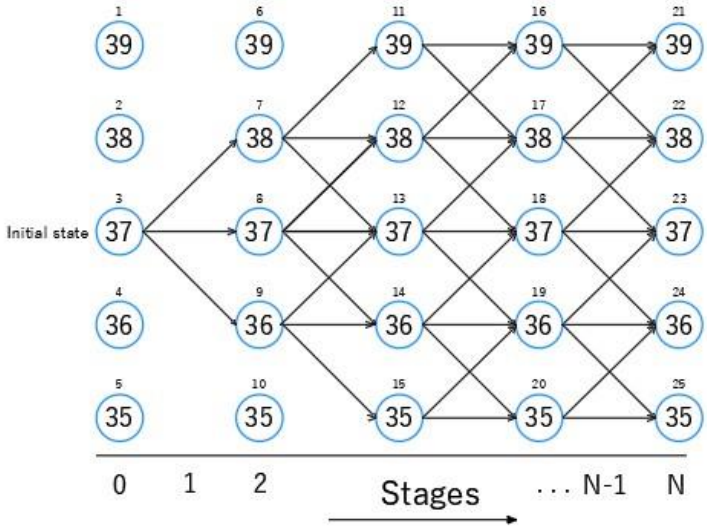


Fig. 59. Weighted acyclic graph composed of temperature states. (Adapted from[25]).

In Fig. 59, each node indicates the temperature state of the water within the tank at a given stage of the simulation. Nodes in the diagram are connected by edges. Each edge represents the cost (in energy and consumer comfort) to move to another temperature state. Dijkstra's algorithm is used to navigate through the graph from the initial to the final state, discovering the most cost-effective (shortest) path on its way.

[32] A directed acyclic graph (DAG) is a form of directed graph (digraph), a set of vertices arranged in a grid-like structure of nodes that are connected by directed edges. In a directed graph, each directed edge connects an ordered pair of vertices [152].

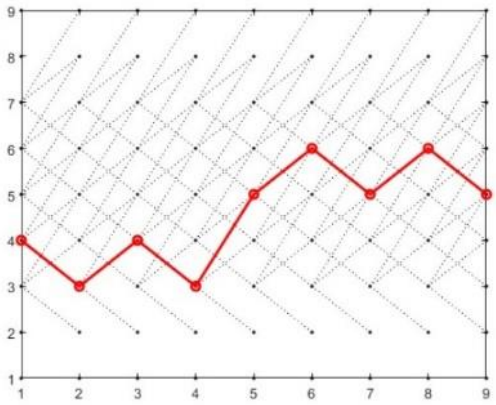


Fig. 60. Dijkstra's algorithm searches between nodes of discretised state space.

Unlike the recursive DP approach used in chapter 7.1, Dijkstra's algorithm separates the search space into sets of visited and unvisited nodes. Visited nodes are ranked in a priority queue according to their perceived *value* in the search space. Value is calculated by the sum of accumulated costs to reach the parent node and the edge cost to reach the next child node.

7.6 Advantages of Dijkstra's Algorithm

An example of a search conducted by Dijkstra's algorithm for a single day is plotted in Fig. 61.

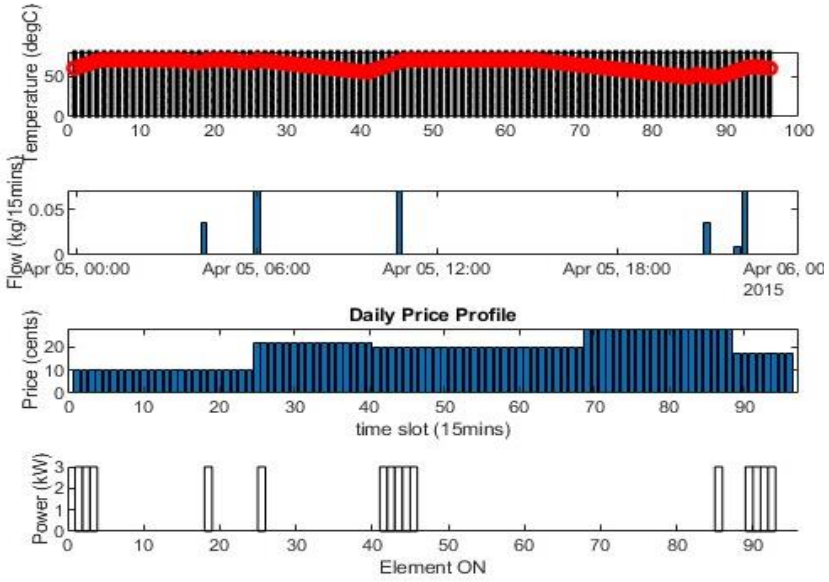


Fig. 61. Dijkstra's algorithm solution – the heating schedule avoids periods of peak demand communicated by higher electricity price signals from the system operator.

Due to the reduced search space used by the algorithm, this approach is at least ten times faster than the DP technique, solving on average within 30 seconds. However, the discretisation of the search space constrains the possible heating and cooling trajectories to ideal rates, which might introduce a large source of error and cause inaccurate estimates. Increasing the number of states to compensate could greatly increase solution times, negating any advantage achieved.

7.7 A* Search

The third dynamic programming technique explored during the project is the A* ('A star') method, which shares the same pedigree as Dijkstra's algorithm and is often used to navigate through 2-D grid spaces common to computer games.

Unlike Dijkstra's algorithm, A* can make an 'informed search' in an 'implicit' search space [53]. This means the A* technique employs 'heuristic' as a simple estimation of the distance to the final state to improve the efficiency of its search. Undertaking the search heuristically also allows A* to discover the search space as it moves, not requiring, as Dijkstra's algorithm did, a discretised map of the search space already in memory to work its way through.

Instead, the A* algorithm must fill in its map of the search space as it explores, deciding based on the perceived value of its measured state and the search heuristic, what the next best step should be to find the shortest path to the goal state.

This application of A* search to the optimal control of hot water heating is demonstrated by Ritchie et al. in [155] and [156].

7.8 A* Search Heuristic

A 'heuristic' employed by the A* algorithm estimates the distance from its current position to the goal state. In a square grid, for example, the distance from point A to point B might be estimated using a line-of-sight measurement from A to B, or the 'Euclidian' distance. [33]

[33] The Euclidean distance is the straight-line distance between two points. It is calculated as the square root of the sum of squared differences between the x and y components of the points in the Euclidean space.

Knowing from the heuristic measurement that the search is more likely to be shorter if certain directions are explored in preference to others allows the A* algorithm to prune the search space it must explore substantially [53].

The A* algorithm breaks the cost of exploring the search space into two parts. There is the cost to get from the initial state to the current state, the 'cost to arrive' g(u). The cost of the heuristic value is described as the 'cost to go', h(u) [53].

The total cost to be at a state in the search space is [53]:

$$f(u) = g(u) + h(u) \tag{7.2}$$

Conceptually, as is shown in Fig. 62, having searched from the initial position x_0, the next best node to select along the shortest path to the goal is x_k, the node with the lowest estimated value h(u).

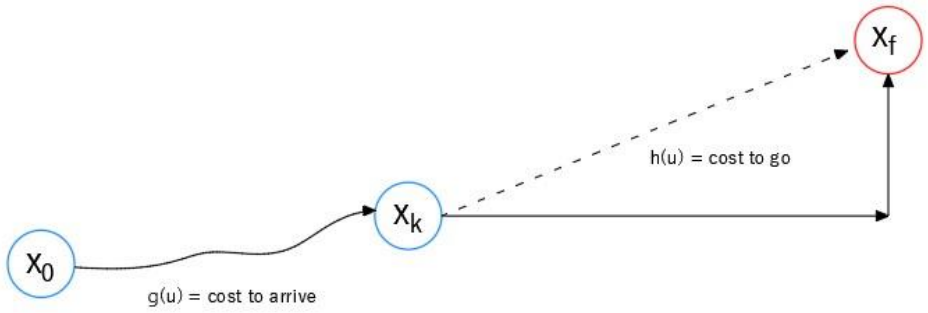


Fig. 62. Using heuristics to find the shortest path.

To calculate h(u), the Euclidean distance represented in the diagram by the broken line is the shortest distance. In practice, to simplify the calculation, 'Manhattan distance' is calculated instead as the sum of the horizontal and vertical components of the straight-line path from x_k to x_f.

7.9 Implementation of A*

The results of implementing the A* algorithm are shown in the plot of Fig. 63. In this case, the set-point temperature of the starting and finishing states has been set to 60 ^{o}C.

The heuristic value h(u) was set to the 'Manhattan Distance' to simplify calculations, but this also has the effect of guiding the algorithm more quickly in the vertical search space to find the setpoint temperature.

Of note is that A* maintains the setpoint temperature at this level even when there has not been a water draw from the tank, which makes it very much like the traditional 'thermostat control' seen in normal hot water cylinders. Discounting the ability of the algorithm to pre-heat the tank in anticipation of a scheduled water draw period, the downside is that the raised tank temperature throughout the remainder of the day causes passive heat loss from the tank.

Passive heat loss is one of the most significant sources of inefficiency in a hot water cylinder. Consequently, the algorithm would not be suitable as a control method in this form.

7.10 Improved Controller using A*

An improved version of the A* algorithm was arrived at by instead requiring the algorithm to aim for a final temperature state at the minimum that would ordinarily be deemed 'comfortable' by a hot water user, around 40^0C.

The algorithm was recoded into a two-node "stratified model" with a reduced time step of 3 minutes. The flow diagram for the code structure used to implement the algorithm is provided in Appendix I. The trade-off of this approach (given the increased size of the search space) was to reduce the scheduling time to four hours. In a practical controller, this would act as a kind of rolling horizon, analogous to the strategy used in Model Predictive Control.

Using this approach, the algorithm can maintain the standing temperature of the tank at the minimum possible level to guarantee user comfort in case of unexpected water use. In anticipation of scheduled water events such as showers and baths, the controller pre-heats the water to match the expected energy that will be lost from the system.

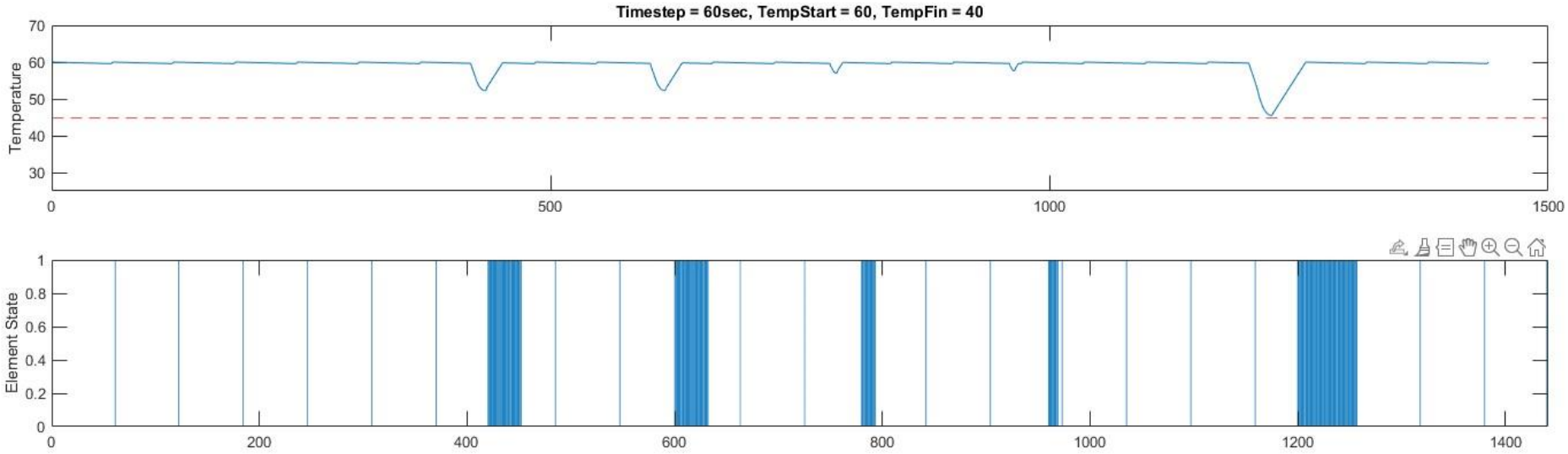


Fig. 63. A heating schedule (single day). Notice that the temperature is maintained at around 60 ^{O}C in anticipation of water usage events. Being maintained at this high temperature relative to the ambient environment introduces a standing loss that is wasteful of energy. The final behaviour observed is similar to thermostatic control.*

An improved heating response is seen in Fig. 64 where the water temperature rises to the exact level required to cover the subsequent water use before the expected demand event.

Further development would be required for this version of the A* algorithm to implemented in a working hot water cylinder. A notable weakness of the algorithm in its present form is that it requires 'perfect foreknowledge' [141] of the timings of future hot water draw events in order to determine appropriate pre-heating times in advance of water use.

A second issue centres around the possible risks of allowing the EWH to remain just above the minimum comfortable temperature of 40 ^{o}C for long periods, possibly allowing for the growth of water borne pathogens such as the bacterium Legionella, which if inhaled can cause a form of pneumonia known as Legionnaires disease [157].

To combat this risk, the algorithm would need to be modified to ensure heating to at least 60 ^{o}C on at least one occasion per day.

The final MATLAB code developed to implement the A* search optimal hot water control algorithm described in this chapter is presented in Appendix J.

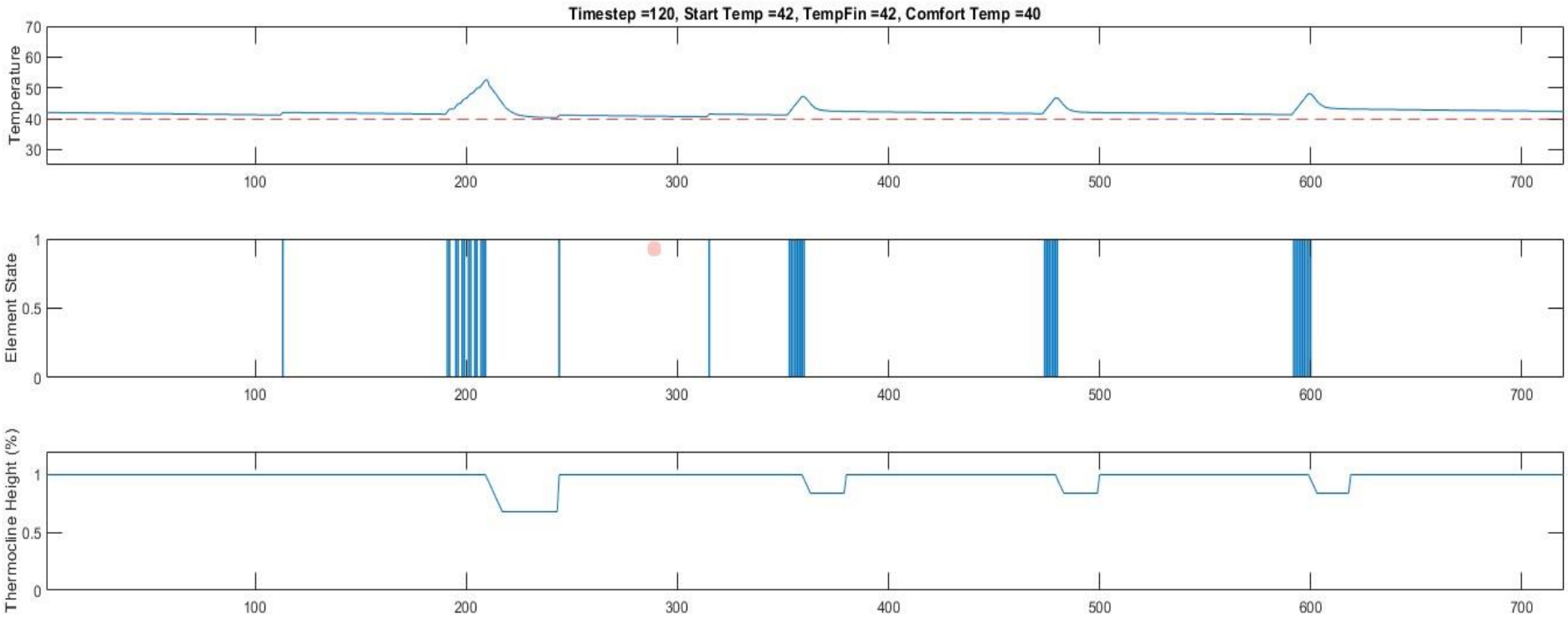


Fig. 64. Improved A featuring optimal control of the heating element. The code for this algorithm is provided in Appendices I&J.*

7.1 Summary

Three different approaches to water scheduling control based on Dynamic Programming algorithms were implemented in MATLAB. All were found to be computationally expensive to operate, and the results were less impressive than some results discussed in the academic literature. At a time step of 900 seconds, the traditional DP algorithm gave reliable results and was used to generate heating schedules in the embedded network simulation to be discussed in chapter 8.

Dijkstra's algorithm offered fast solution times because of a discretised search space enforced on the algorithm. The nature of the discretization, however, seemed to reduce the reliability of the hot-water model it was simulating, since the heating and cooling profiles of the hot water were sometimes constrained along unnatural heating and cooling rates.

The A* algorithm, on the other hand, was not limited in this way and could reliably find solutions to the heating schedule but sat at the set-point temperature, incurring high passive losses.

An improved version of the A* algorithm modelling a more accurate stratified hot water cylinder model over shorter time periods overcame these difficulties. With some modifications, including the ability to predict hot water use events and extra heating to avoid the growth of pathogens such as Legionella, it could form the basis for developing a controller developed to operate on a real world hot-water cylinder.

Chapter 8 Model Description

To answer the question of whether optimised hot water control can act as a useful DER resource to reduce peak loading and shift excess consumption, a model community of houses connected on an embedded network was assembled using MATLAB/Simulink.

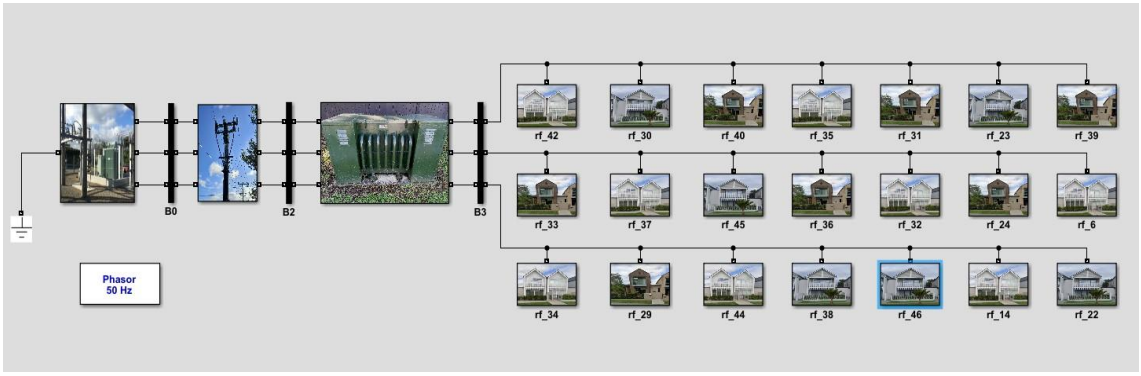


Fig. 65. Community microgrid simulation interface in MATLAB/Simulink.

The model, as shown in Fig. 65, is based on consumption data from 21 of the households obtained from GREEN Grid house electricity consumption profiles described in chapter 3. For this study, the houses in the network are connected to a single transformer to determine the aggregate load on the grid under differing circumstances. Each house has the added ability to connect to hypothetical photovoltaic (PV) systems to simulate household PV generation. The houses are also included with a simulated EV charging system, to explore the ways in which hot water load control might complement increased penetrations of EV ownership in small communities in the future.

This chapter will describe the model and discuss the design and functioning of each household unit it contains.

8.1 Benefits of Embedded Networks

So far in New Zealand, there are relatively few successful community distributed energy resource (DER) projects in existence [158]. In comparison, traditional Embedded Networks do not have a high profile but are a surprisingly popular arrangement, with at least 10,000 examples operational across the country as of 2015 [158].

The regulations and legal protections surrounding embedded networks however may soon prove an important vehicle in the emergence of DER and microgrids. Small networks of connected households each providing energy to the collective through installed solar or batteries have the potential to reduce grid load at crucial times of heavy demand.

Connecting to the grid through a single network owner/distributor also creates an opportunity for the small-scale DER of these communities to be aggregated, forming a critical mass of load for demand-side management ancillary services, and creating the knowledge and experience for a new kind of flexibility provider to develop and enter the New Zealand wholesale energy markets [159].

Embedded networks however are likely to increase in popularity for several other reasons, as DER technology makes its way into the mainstream of the energy industry. The potential for embedded networks to share locally produced energy with each other creates many possibilities, not the least reducing the demand that communities in new developments might place on the grid compared to traditional housing arrangements. A select group of houses forming an embedded network that could generate a large share of its own electricity, could conceivably use a smaller connection point to the grid with much reduced supporting infrastructure costs than a group of houses not sharing a residual ICP connection.

Still other communities with growing electrical demand could potentially defer the costs of network upgrades if the energy generated and consumed within their community could be shifted throughout the day to minimise peak loading. This would also result in lower costs to the network owner and local EDBs. The reduced costs in network development and maintenance are likely to be significant. The benefits are likely to trickle through to electricity end-users as well.

8.2 Overview of Embedded Networks

In the early history of electrification, before the national electricity transmission grid was established, smaller local electrical networks could connect to each other to ensure greater reliability and share costs [160]. A secondary network is now defined as an

electricity network that is “indirectly connected to New Zealand’s national electricity transmission grid” [161].

In contrast, an *embedded network* is an electrical network that connects to the grid through a single installation control point (ICP) controlled by the network owner who acts as the distributor for the network. Consumers within the embedded network have their own ICPs and may decide which electricity retailer to purchase from [161].

Embedded networks are very similar to microgrids [161] in that they can “operate independently or in conjunction with the area’s main electricity network [161].”

However, embedded networks occupy a slightly different niche in the electricity market in comparison to the other forms of secondary network. They differ from customer networks in that all consumers in the network have their own ICP and the freedom to choose their own electricity retailer [161].

Unlike a network extension[34], an embedded network is not connected directly to the grid, and all electricity charges incurred within the network are subject to reconciliation with the network owner to determine the amount of electricity purchased [162].

Even within seemingly ‘typical’ communities, there is an enormous variation of electricity demand between households. This is because of the differing levels of occupancy, occupant activity and lifestyle habits of ‘typical’ residents.

Though not nationally representative, the households of the GREEN Grid dataset still contain many patterns typical of New Zealand electricity consumers that could not be captured in generic or probabilistic models.

Modelling the network in the MATLAB/Simulink environment allows some flexibility around the use and manipulation of the original GREEN Grid dataset, as streams of authentic data can be mixed with simulated energy profiles to explore various scenarios of hypothetical energy use.

[34] A network extension is a category of secondary network, typically comprising office buildings and residential apartment complexes, that provide (own) the network infrastructure [161].

8.3 Microgrid Simulation Model

The embedded network community simulated in this model is composed of a basic network of 21 identical household units. The network includes an 11 kV three phase medium voltage power supply, connected through transmission lines to an 11 kV/400 V step down residential transformer, as depicted in Fig. 65.

Each of the houses is connected in random order to one of the three phases of the network. Each household model's input includes GREEN-Grid data for general household circuits, separate hot-water electrical consumption and related NIWA weather files. A two-node 'stratified' hot water cylinder is included for those houses in the GREEN Grid survey with standalone electric hot water cylinders.

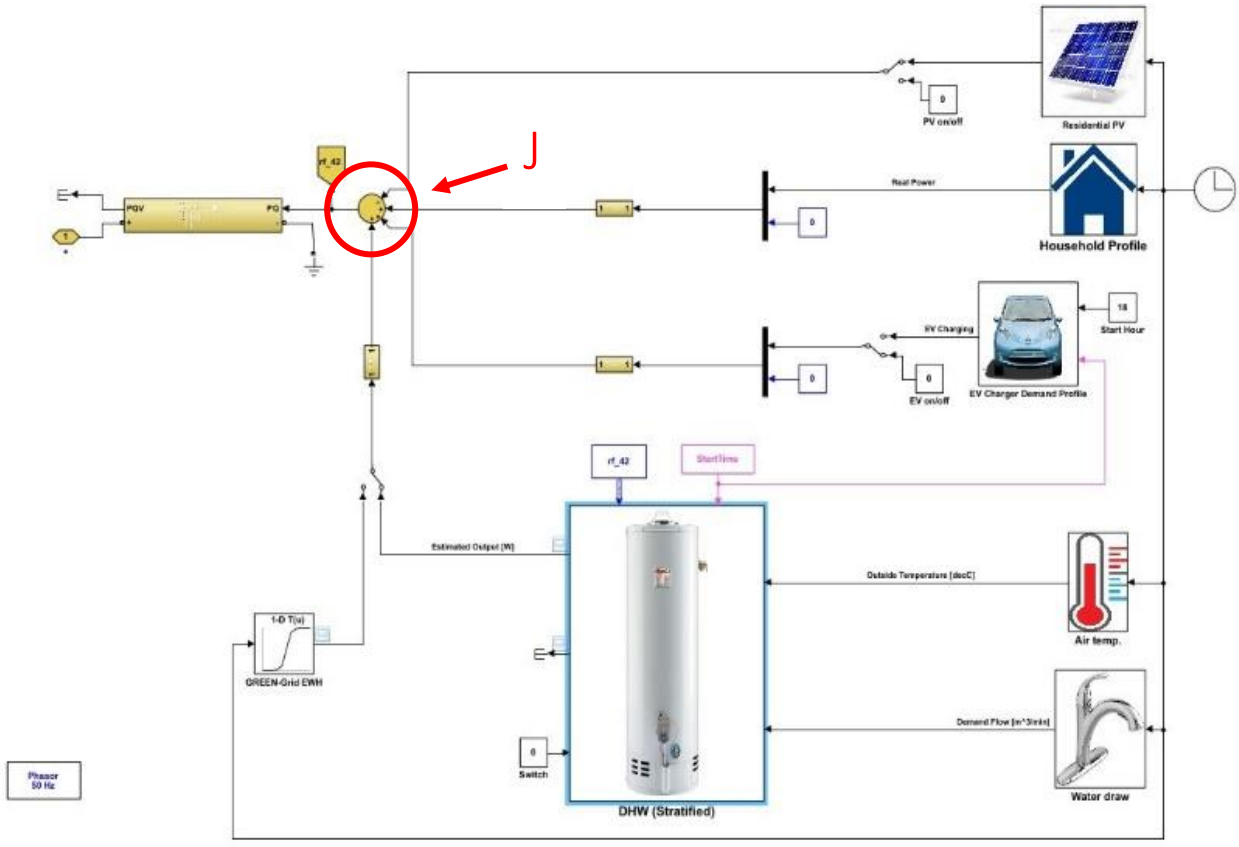


Fig. 66. Typical household model from the community depicted in Fig. 50, showing the main appliances simulated alongside electric hot water. (Appendix K)

The model is supplemented with simple models for an EV charger and residential PV system. The output and draw of each individual circuit is summed at the junction

labelled “J“. The net load is output from the house to be aggregated with the load profiles of the other houses in the community.

Toggle switches are interspersed throughout the model to allow components to be switched on and off when necessary. These include the residential PV and EV charger. Similarly, a toggle switch makes it possible to select between running synthesised or historic GREEN Grid hot water data.

8.3.1 Electric Hot Water Cylinder

Depending on the degree of complexity required, the electric hot water cylinder in the simulation can be modelled in various ways, each with increasing levels of sophistication and accuracy. In this project, the two-node lumped parameter models discussed in chapter 5.4.1 were implemented in MATLAB/Simulink.

A Simulink implementation of the fully mixed DHW model, based on the derivation of [70], is shown in Fig. 67.

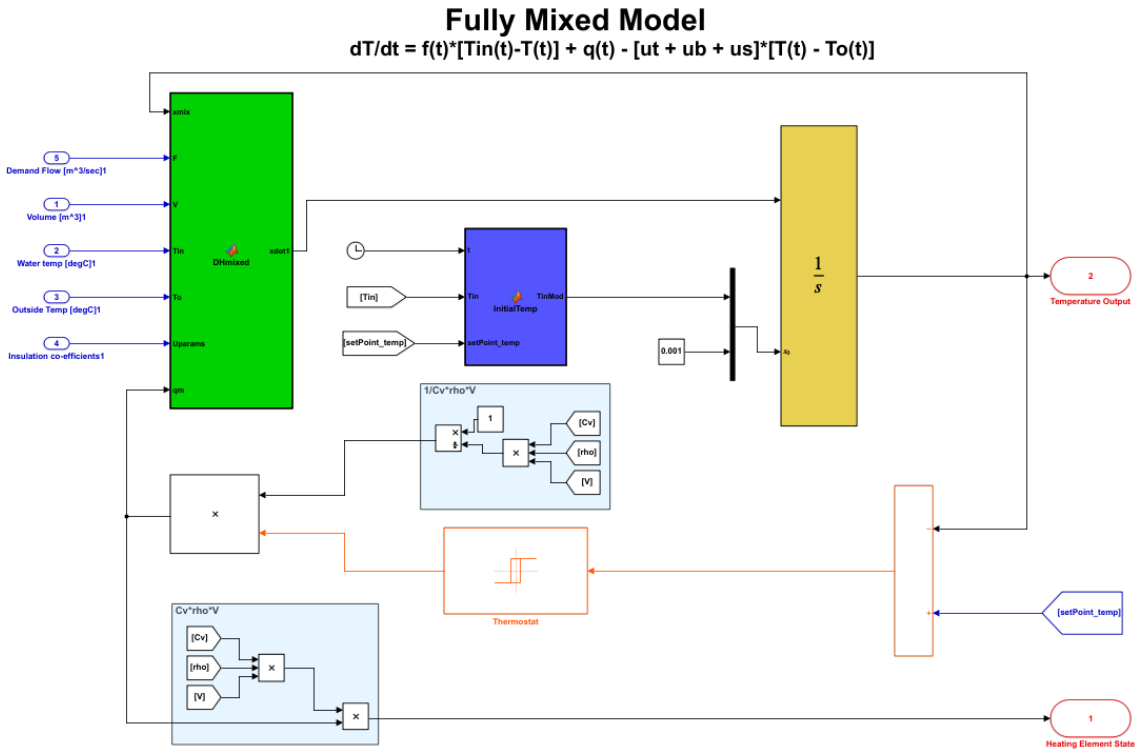


Fig. 67. Fully mixed block diagram code of an EWH model in Simulink. The block in green contains the individual ODE equations determining the behaviour of the system. The orange block is an integrator.

In this implementation, the main ODE system equations are stored in the green function block on the left-hand side of the block diagram. Integration of the ODE is carried out by the yellow block on the right-hand side of the Fig. 67.

When the system is at rest, the water in the tank is assumed to be 'fully mixed', with no stratification present. When hot water is drawn from the tank, a new state is activated, and the model jumps to a system of two differential equations that describe the relative volumes of heated and unheated water in the tank.

For this reason, the implementation of a two-node stratified hot water cylinder is treated as a state machine as depicted in Fig. 68 and described in chapter 6.4.2.

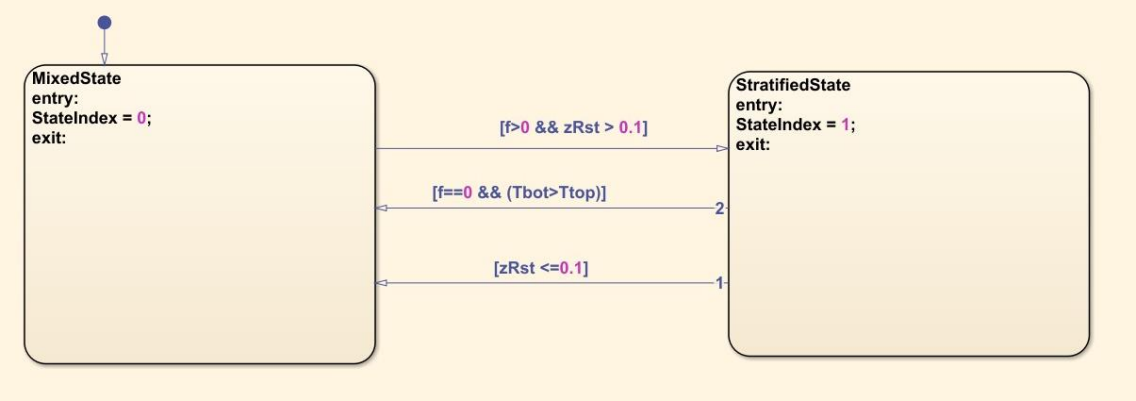


Fig. 68. Stratified model depicted as a state-machine in Simulink Stateflow.[35] The state machine will change state when the logical conditions displayed on the central arrows are met.

8.3.2 Electric Vehicle (EV) Wall Charger

To incorporate the effects that increased EV ownership is likely to have on network peak demand profiles, and the degree to which managed hot water control might alleviate those effects, a simple model of an EV wall charger was included for each household.

The model allows the charging rate to be modified according to the specific vehicle and charging rate envisaged. For example, the wall-plug charging of a Nissan Leaf could be simulated by adjusting the output gain (orange triangle in Fig. 69) to 3300 W.

[35] Stateflow is a graphical language interface that models state transitions to events and input signals in the MATLAB/Simulink coding environment.

The model provides for a likely charging start time to be entered before simulation. Random number generators add variance to this value according to a gaussian distribution profile. Similarly, the minimum charging time is set to 90 minutes. Still, it will randomly vary by up to four hours in keeping with estimates for typical New Zealand driving behaviour estimates obtained from [85], [163].

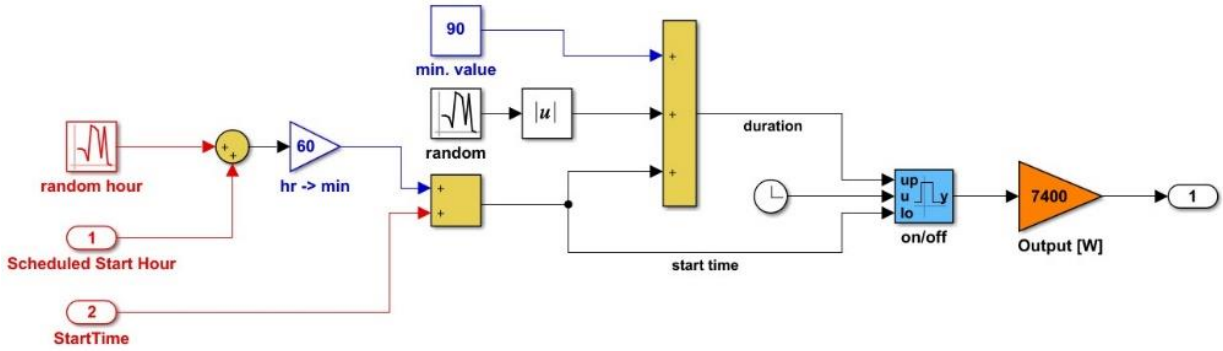


Fig. 69. Simulated semi-random turn-on/turn-off model of an EV wall charger. A constant charging rate is set by modifying the value in orange at output 1.

8.3.3 Coordinated Charging for EVs in Network

To prevent the presence of EV charging from overloading the network at peak times and overwhelming the model's visibility of the effects of optimised hot water control, a rudimentary scheme of charge control co-ordination was implemented using unit-commitment integer linear programming.

A comprehensive study of the benefits of co-ordinated EV charge scheduling would require study in its own right and is beyond the scope of this thesis. The topic is briefly addressed here to illustrate the potential impact that widespread EVs adoption is likely to have on low-voltage distribution lines in the future.

Adding a simple scheme in this simulation allows for simulated EV charging periods to be spread across periods of low consumption, particularly the early morning hours to prevent the significant peaks caused by the higher EV load distorting the effects of reduced peak hot water load on the network.

The code employed is available for inspection in Appendix L.

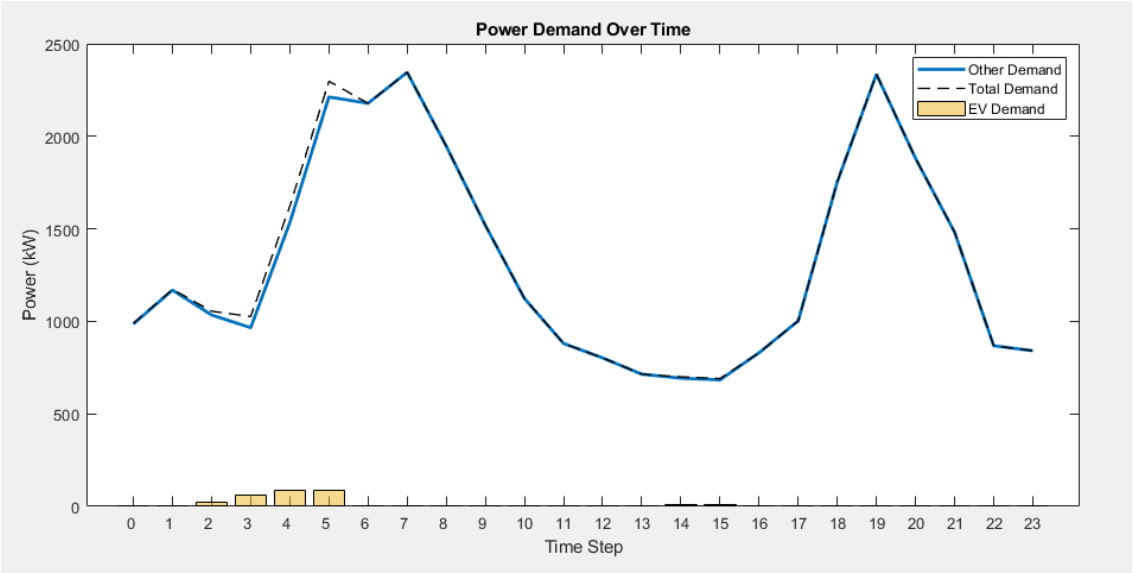


Fig. 70. Scheduled overnight EV charging and its effect on the overall power consumption profile of the microgrid.

8.3.4 Photovoltaic (PV) System

The simulated PV system uses NIWA insolation weather data obtained for the New Plymouth and Hawkes Bay regions during the period the GREEN Grid project was active.

For the data available, a gain value of 6000 approximates a typical small-sized residential system of approximately 3.5 kW[36]. A saturation block is added to the model to ensure the output never exceeds this value, simulating the ceiling output of a PV inverter.

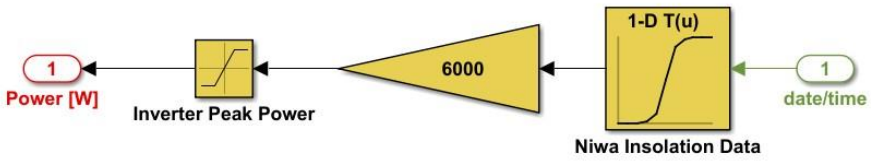


Fig. 71. Solar PV model - Insolation data is modified by a gain value to approximate system output.

[36] The average residential PV system in New Zealand in 2020 was around 3.5 kW [164].

8.4 Simulation Results

The simulation results of the model embedded network simulation are based on two main scenarios involving the regular thermostatic control of hot water cylinders and the case in which scheduled hot water control is applied to houses fitted with optimised electric hot water cylinders.

8.4.1 Selection of Timeframe

To best illustrate the effectiveness of scheduled hot water to reduce peak demand by shifting load, the colder winter months were chosen to be the focus of the first simulations.

The dataset from the GREEN Grid project was the most complete for the group of 21 households selected in June. The heaviest demand period in June can be seen in Fig. *72* as occurring in the last working week of the month, between 22-26 July. Weekend days were excluded from the simulation as the peak demand was reduced and activities spread more evenly throughout the day [21].

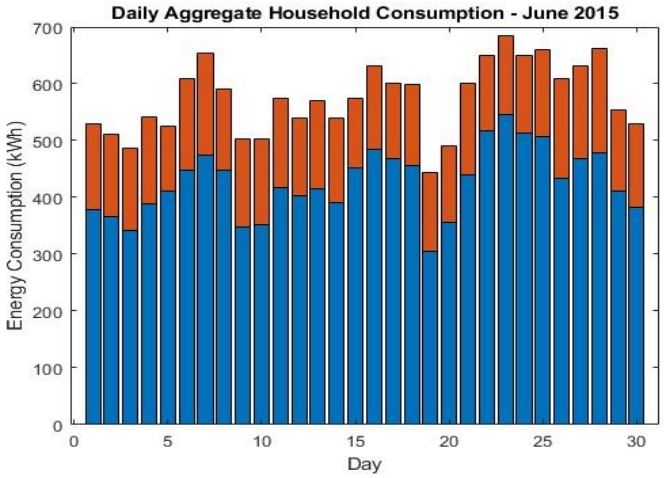


Fig. 72. Aggregate Energy Demand for June 2015. (Blue = appliances, Orange = Hot water).

8.4.2 Aggregate Household Demand Profile

Due to data limitations in the original dataset, of the 21 houses used for the model, 19 are equipped with electric hot water cylinders, 1 with a wetback and 1 with a hot water heat pump. The aggregate demand profile of these houses for one week is plotted in *Fig. 73*. Interestingly, the month of June is recorded as having especially unsettled weather,

with a large storm beginning on 22 June that caused flooding in some parts of the Taranaki [165].

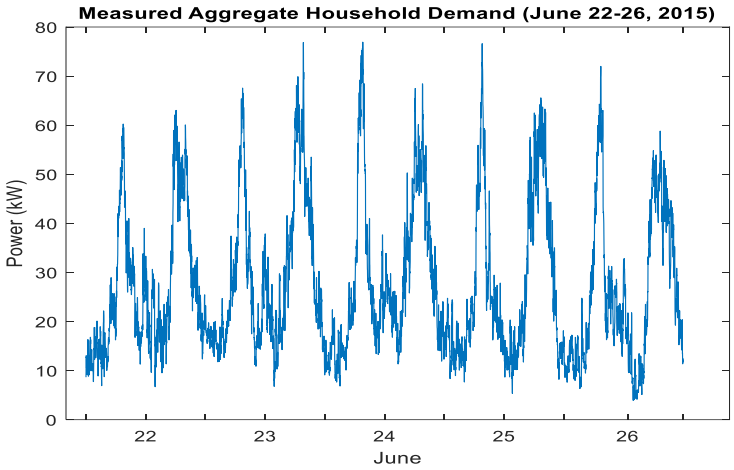


Fig. 73. Aggregate Household Power Demand (22 - 26 June 2015).

The peak aggregate demand for the period ranges from 60 kW-78 kW, and are also visible in the tail of the histogram of Figure 74. Possibly due to the stormy conditions at the time, the days from Wednesday to Friday feature morning peaks greater than evening peaks.

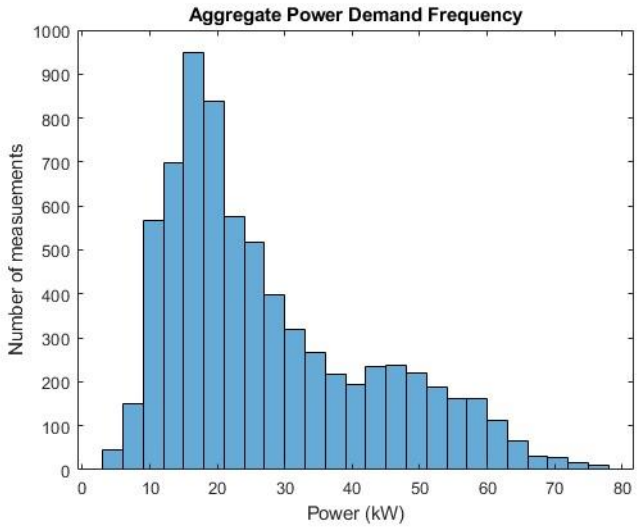


Figure 74. Histogram of aggregate power demand for 21 houses during June 2015.

8.4.3 Comparison of Baseline and Controlled Scenarios

Application of the scheduled hot water profiles to the applicable houses is applied in Fig. 76 to show the difference caused by the peak load profile across the five days of the working week. The original baseline profile is shown in blue, and the new profile with hot-water heating loads shifted out of the peak periods is superimposed in red.

Shifting the peak hot-water heating load has reduced the peaks significantly but has caused new mid-day peaks to appear where there were none before. This is partly the result of the long 15-minute step used by the DP algorithm and the reduced number of options allowing heating to be scheduled. The main cause, however, is the lack of coordination between hot-water cylinders, an important aspect of the problem that needs to be developed in a real-world solution.

Even allowing for these deficiencies in the model, an effect is created, and the reduced demand at peak moments is of the order of 20 kW in extreme cases. This is clearer in the histogram representations of *Fig. 75* where the largest peak of the controlled scenario is 62 kW, well short of the 78 kW peak of the uncontrolled scenario. Even so, it still represents a reduction of the order of 20% over the worst case during this period.

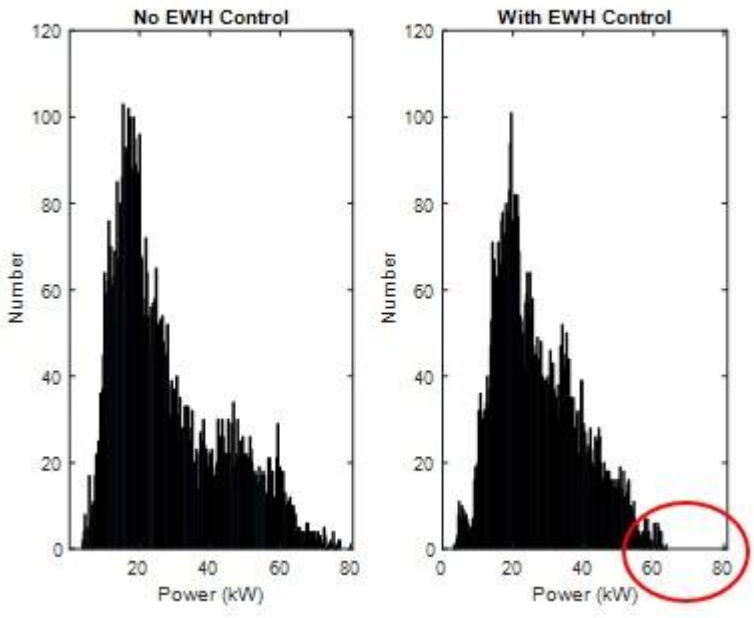


Fig. 75. Histogram comparison of peak demand scenarios showing clear reduction at power levels > 60kW (absent inside red circle).

8.5 Extent of Benefit

The extent of the benefit to stakeholders is difficult to establish based on this limited dataset. Anecdotally, in the sense that the period simulated, a heavy storm event during the week of the winter solstice is likely to be a period of significant stress on the grid.

The question is important due to the significant levels of extra network investment that peak demand growth is likely to cause – as much as $220 more per household per year by 2050, according to one recent estimate [166].

In a near worst-case scenario, a rudimentary hot-water control system using personalised scheduling has been shown to reduce the peak demand load on this small section of the network by up to 15-20 kW for some hours of the day.

Granted that ripple control might also be expected to operate at such an extreme time of the year, the reductions in this case will have occurred without any noticeable decrease in consumer comfort. In contrast to ripple control, which is activated intermittently, the benefits in peak reduction obtained by a semi-autonomous system of individual hot water cylinders would continue throughout the year instead of only during extreme seasonal events as with the current regime.

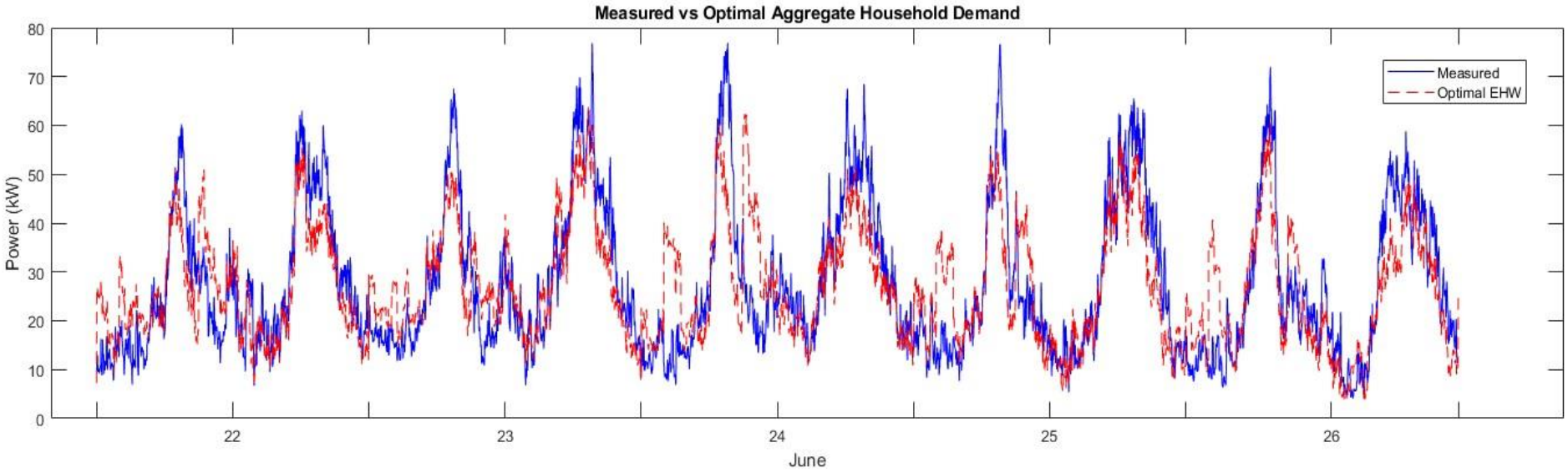


Fig. 76. Comparison of controlled and uncontrolled hot water load profile effect on aggregated load for five days.

Various figures have been used in the electricity sector to estimate the savings in avoided peak loads. These range between $100 – 240 /kW/year [14], [18], [167]. The savings are made up of three relatively evenly distributed components: transmission, distribution and renewable peaking generation [18].

Using the same line of reasoning as Mair et al. to estimate the peak demand potential of home batteries [100], the peak demand savings of controlled hot water can be reasonably substantial, even for a small community network such as this.

Assuming an avoided peak load value between the above Fig., the demand saving could be $1500 and $2500 per year in deferred network improvement costs. If the embedded network was a new development, a smaller transformer might also be installed than would otherwise have been planned to also help with the upfront reduction of costs.

A further saving to the EDB would be avoiding costs associated with maintaining a ripple control system for this group of houses, which is not insignificant and is estimated annually at $10-$29 per ICP [14].

To the embedded network owner, savings will likely depend on the nature of the contract established with the relevant EDB and tenants using the embedded network. For a network of this size, aggregation of reserve loads is unlikely. Still, for larger communities, the opportunity could exist to offer reserve directly into the wholesale market as a fast interruptible load.

The benefit to the end user consumers of hot water is likely to be more modest and, to some degree, will be present in reduced lines charges due to the reduction in peak load. However, direct energy savings are still likely to be modest since energy is still required to heat water and maintain its temperature.

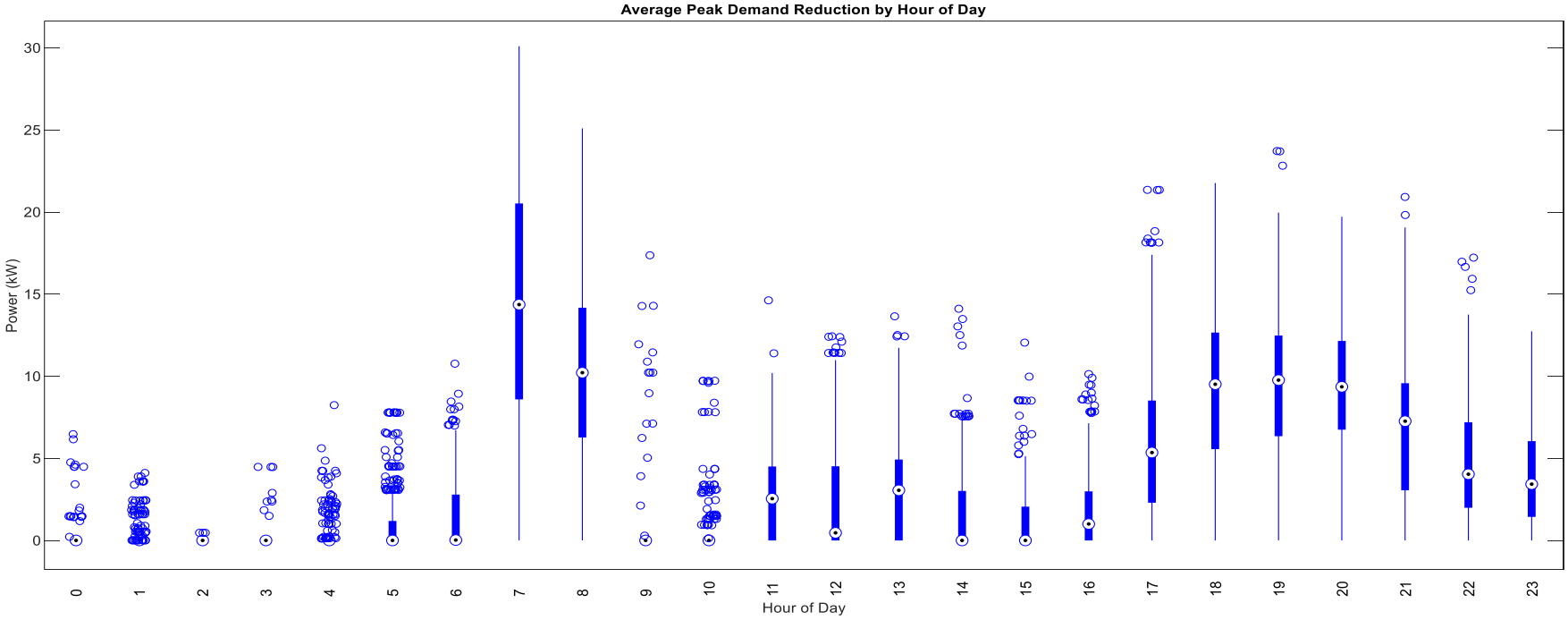


Fig. 77. Average hourly peak demand reduction for simulated period.

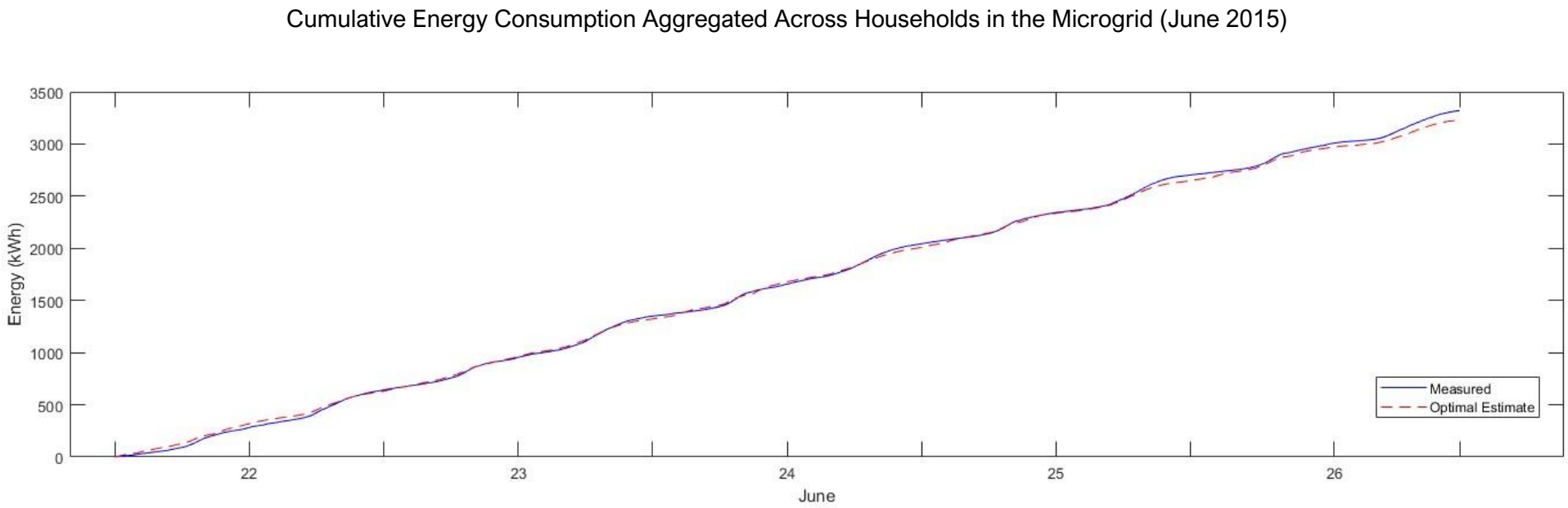


Fig. 78. Aggregate energy consumption for June (kWh).

As shown in Fig. 78, the energy savings in the aggregate over the month are minor but not insignificant. Where stakeholders are most likely to see benefit is in a reduction in peak demand charging depending on the nature of the tariff structure offered by their local EDB. In the case of all houses following a day/night tariff with six peak/off-peak times. [37]

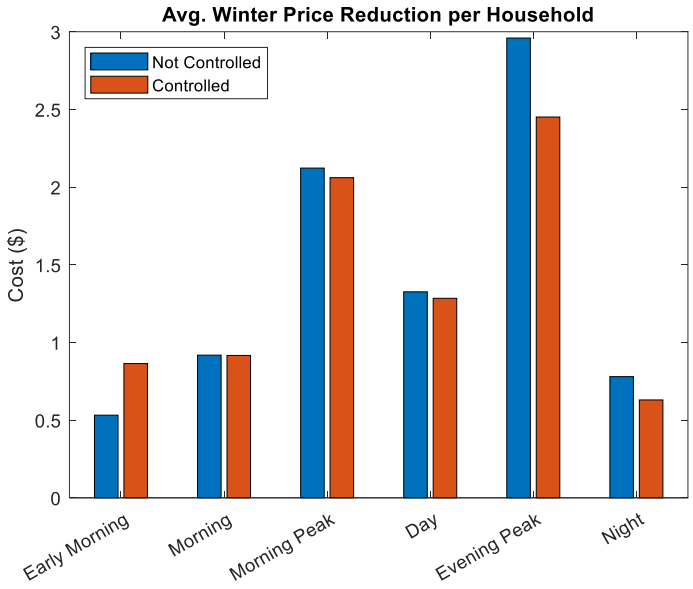


Fig. 79. Average price paid per household across tariff time zones in June 2015.

Moving hot water demand out of the peak morning and evening periods reduced the average electricity bill paid per household by approximately $2.50/week, thanks to the lower price charged for off-peak consumption.

8.6 Summary

Simulation of the embedded network over a period of annual peak consumption in June suggests that in an embedded network utilising personalised scheduled hot-water control, modest benefits might be realised by all stakeholders: the connected EDB, the embedded network owner and consumers connected on the network.

The main benefit is observed in a reduction of up to 20% at certain times of the day of peak demand on the network. This reduction was most clearly seen during evening and morning peaks.

[37] Based on the Ecotricity RVT_ARHL_sol_002 price plan.

It represents a gain to all parties in the form of reduced lines hardening and reinforcement costs that could save the EDB as much as $2500 per annum in deferred costs.

Chapter 9 Research Findings

This research began with the aim of understanding what alternatives there are to thermostatic and timer control of residential hot water cylinders. Having identified several prominent approaches, the study also aimed to determine which of these was the simplest and most easily implemented in the future, as New Zealand's electricity grid adapts to challenges of accelerated electrification.

Delving into the topic has shown it to be larger than anticipated, because it involves several disciplines, from thermodynamics to computer science to the workings of the electrical grid. The effects of hot-water control are also likely to be felt at multiple levels of the electricity system – from the level of the individual household to local electricity networks, through to the national grid and wholesale electricity market.

Overall, some simple conclusions have been made which are summarised here, based on the work carried out to date, which also suggest where further work could be undertaken in the future.

9.1 Simplest hot water cylinder model

Of the several types of model that were examined, the most effective hot water cylinder model was judged on joint criteria of simplicity and accuracy. Single-node, fully-mixed models were found to be overly simplistic, and because they are unable to assess stratification within the tank, are not able to accurately predict basic characteristics of hot-water cylinder operation.

More complex 2-D and 3-D models were not implemented, but their complexity requires too much detailed understanding of fluid dynamics and complex programming to be practical. Of the other 1-D models simulated in code and compared to laboratory results, increased complexity did not noticeably improve model simulation performance.

Comparing the multi-node models studied in this research, the two-node, stratified performed as well as the more detailed multi-node models tested in predicting the outlet water temperature. As this model was also able to track the height of the main-

thermocline, it is easily adapted to predicting user comfort and estimating the overall state of charge of the system. Because the code of this model is less complex, it could also be more easily installed on simple embedded systems.

9.2 Benefits of Complexity in Hot Water Cylinder Control

There were two aspects to the complexity of hot water cylinder control models that were encountered during the course of this research. These were not well understood at the beginning when the research questions were originally formulated.

The first is the complexity of the general architecture and design of the controller. Adequately modelling the internal dynamics of a hot water cylinder is important because it is on this basis that command signals to the heating element will be executed.

There are diminishing returns however in modelling the system in too much detail.

Firstly, the internal dynamics are complex enough that only a very detailed and computationally expensive model could capture all phenomena surrounding the interactions between heat and water inside the tank.

Secondly, hot water cylinders have individual aspects of their design, which make them inefficient in many differing ways. These deficiencies are unlikely to be captured without a highly customised and complex modelling process that could not be easily generalised in practice.

For instance, the hot water cylinder tested in chapter 4 had been designed without insulation to the pressure valve, which acted as a thermal bridge and reduced thermal efficiency of the entire hot water cylinder. A more complex simulation model could have been developed to address losses of this kind. The added variable would then have required tuning, which would increase the overall accuracy, but would not necessarily make the model's predictions overwhelmingly better.

In some cases however, more detailed knowledge of the state of the tank might be beneficial. For example, the multi-node models that estimate the heat at various levels

inside the tank could be useful, by providing options on where to install temperature sensors in retrofitted or newly designed tanks with temperature sensors.

For example, during the experimental stage of this research, it was found that the lower section of the hot water inflow pipe at the base of the hot water cylinder, closely tracked the temperature on the main body of the hot water tank. A multi-node model that could estimate temperatures based on samples from sensors in this position might be effective in predicting outlet water temperature from levels higher in the water column.

Using this insight, the placement of a single sensor around the inflow pipe could be far more cost-effective and accurate than attaching multiple sensors to the outside of the wall of the tank. It would also make installation of thermocouple sensors more practical to achieve, especially on older hot water cylinders requiring retrofitting of the controller device.

9.3 Advantages of Simple Algorithmic Control

In experiments with the small community model in Simulink, one of the main effects of simple timer control was to remove diversity from the system. Many hot water cylinders programmed to activate at the same times of the day in preparation for say morning showers, could cause large spikes in demand around common peak period times.

At the small community/embedded network level this is less a problem than at a regional level. Algorithmic control on the other hand can help maintain some diversity in local networks by responding to the random variability still present in the states of individual hot water cylinders.

The chief advantage of a distributed algorithmic form of hot water cylinder control would be to shift hot water heating loads outside of peak period times without the need for intervention from a central authority or system operator.

Allowing individual cylinders to operate at lower temperatures when not anticipating demand, could also help to reduce standing losses in individual hot water cylinders. Cumulatively, this could add up to a large energy saving, as long as water temperatures were allowed to peak at least once daily to prevent the growth of legionnaires disease and mitigate that risk.

Invariably, as the penetration of EVs across households increases, the effects of battery charging will also need to be co-ordinated. This will be easier to integrate into household energy management systems if there is already an autonomous means of controlling electric hot water.

9.4 Advantages of Shortest Path Search as an Algorithmic Control

The algorithmic control method most thoroughly investigated in this research has been a sub-set of the shortest path search methods, namely Dijkstra's algorithm and A* search.

The main advantage of these approaches is their relative simplicity. Both algorithms can find a near optimal heating control strategy for a full day of hot water usage if the exact times of hot water demand are known in advance.

Both methods are also able to track changes in electricity tariff pricing and could therefore react to dynamic pricing signals passed on from the system operator if such a mechanism existed.

Selective heating of hot water loads, in response to external pricing signals and domestic hot water demand patterns could still preserve the diversity of the heating demand seen between different households. The close tracking of the price signal seen in the Dijkstra model simulated in chapter 7.6, suggests that this diversity could be lost if all the houses in a neighbourhood or small community reacted to the same price signal.

The A* algorithm developed and improved in chapter 7.10 was found to be less susceptible to this problem, and by holding water in the tank above a preset minimum temperature for extended periods through the day, could also help to limit standing losses from unused heat leaking from inefficient cylinders.

These benefits however cannot disguise the problem that these algorithms are still single-objective optimisations. Further research will be required to improve them and make them truly multi-objective, and capable of balancing several competing objectives at the same time.

The similarities between the lower-level workings of both architectures, and the fundamental principles of RL algorithms have not been discussed in this research but this could a natural path to follow.

RL algorithms, particularly more modern deep learning types can balance multiple objectives and can adapt to dynamic environments in ways not possible using the simpler Dijkstra and A* search methods that have been discussed.

Further research with shortest path search algorithms that fully explored other variations of this basic strategy could provide useful improvements that address the shortcomings of the algorithms introduced to this point.

9.5 Benefits to Stakeholders

Embedded networks are already well established in New Zealand but usually take the form of large commercial buildings with multiple tenants, such as shopping malls. Instead, the study has presented an embedded network as a community of 21 houses sharing a small distribution network owned by a person other than the EDB in control of the greater network.

Having only a single ICP connection to the greater electricity network and sharing many attributes of a microgrid, this sort of community-focused housing arrangement is expected to become more prevalent in the future. The question of to whom the benefits of reduced peak consumption will accrue remains to be answered in practice, but based on the findings in this study, they are partially spread between the three main stakeholders of such a network.

9.5.1 Benefits to Electricity Distribution Businesses (EDB) and the Greater Network

A reduction in peak demand of 20 kW on such a small network suggests several ongoing cost reductions could be seen by EDBs connected to embedded networks and microgrids of this type in the future.

Assuming the area was a new development, a forecast reduction of this size would justify a reduction in design parameters for the network in terms of the size of the transformer connecting the site to the grid and associated cabling.

Infrastructure already built at the site would be under less strain due to the lower daily peak, estimated to save an EDB up to $2500 per annum in avoided peak costs.

9.5.2 Benefits to Embedded Network Owner

In this case, the benefits to the embedded network owner are not immediate. Still, they are more likely to develop with the increased opportunity to participate in flexible markets in New Zealand. Larger commercial and industrial entities already benefit from the opportunity to aggregate load and sell it as fast interruptible load. Still, the benefits may not be worth the effort for smaller flexibility providers [4]. This looks to change as new flexibility markets allow a new type of flexibility operator to aggregate flexible load and sell back to the network as ancillary network services.

9.5.3 Benefits to Consumers

If only the cost savings on monthly energy bills are considered, a modest benefit is seen to the end consumer. The end consumer will see minor improvements in energy usage, but their water still needs to be heated. Where savings are made, these result from shifting heating out of peak use times where electricity costs are higher. The houses in this study benefited in this way, on average, $2.50 per week using a generic off-peak tariff plan.

However, the actual savings to consumers come via savings likely to be passed on from the system and network operator in deferred costs because of reduced peak demand.

9.5.4 Extent of Peak Demand Reduction

Even on a relatively small network of only 21 households, shifting the heating load for hot water out of the evening peak period resulted in a 20% drop in the peak aggregate demand. Similarly, a reduction of the morning peak approaching 15% was seen.

9.5.5 Sensitivity to Rebound Effect

A common effect in static systems is for new demand peaks to appear in the aggregate power demand profile as hot water cylinders emerge from prolonged periods of direct control. In these scenarios, the synchronised behaviour of multiple hot-water elements switching on can, in some cases, cause peaks larger than the one the hot-water control had sought to avoid.

A significant effect observed during the 21-household microgrid simulation was the magnitude of this effect, especially with all hot water cylinders responding to the same off-peak tariff rates. In the case that only one tariff schedule was available for all customers on the same network, the optimal scheduling algorithms would fit all heating profiles to this same pattern. This would result in significant new peaks forming outside the prior peak demand period. Even on a small network, the large peaks potentially caused by the 'pick-up' effect would need to be managed carefully, highlighting the potential need for an 'aggregator' to enable coordination services at a manageable scale within the embedded network.

9.5.6 Effects of EV Inclusion

A concerning consequence of the rapid electrification of transport is the likelihood that greater EV adoption is expected to increase households' electricity demand by around 33% [87]. Across large populations, diversity effects flatten the peakiness of this new demand source so that it could be that the after-diversity peak seen amounts to no more than 0.5 kW to 0.8 kW [87].

At lower population sizes, such as the one in this study, the contribution of individual EVs to peak demand can be substantial. EV adoption levels of only 50% saw the peak demand capacity nearly double to 120 kW when uncontrolled EV charging was simulated.

In these scenarios, adopting the optimal heating schedules for each house enabled as many as 30% of households to charge their vehicles during the evening peak without crossing the original baseline demand peak.

Chapter 10 Future Work

A key challenge facing New Zealand society is how to fully integrate intermittent renewable energy technologies efficiently into the national electricity grid. Doing so would make energy use in this country cleaner, cheaper and arguably more equitable for all New Zealanders.

This thesis represents one step in a longer journey to explore this possibility. It has covered several topics that each deserve detailed attention and could in themselves be worthy of further study.

Of these, the main areas that this study have identified for further work are summarised:

10.1 Further Hot-water Cylinder Experimentation

The work done on hot water cylinder experimentation was conducted within a short period of time that limited scope of the testing carried out. Future work should look at partially automating the control of the flow valve and heating element. Controlling the flow of water through the cylinder would allow typical daily flow profiles to be simulated over many days and weeks. The data from these tests could be used to further evaluate hot water cylinder models, improve control algorithms, and explore other methods of control such as RL and MLC approaches.

10.2 Further EHW Model Development

Good progress was made in developing the hot water cylinders models and validating their accuracy. Further work should be undertaken to improve the performance of these models and their ability to predict future states of the hot water cylinder. Further work could develop the multi-node model to include a method to track thermocline height the cylinder's state of charge would allow for better predictions of user comfort. Work also needs to be done in adapting the models to work on embedded systems, and to allow the models to be incorporated into a controller or agent.

10.3 Algorithmic Development and Refinement

This research investigated two main types of shortest path search algorithm but would benefit from a further exploration of other common algorithms, including depth and breadth first search algorithms.

The implementation of the A* algorithm in the context of heating hot water is unorthodox, because these approaches are most commonly applied to problems with static state spaces where the landscape for the search does not change.

No other examples of its use in temporally changing environments could be found outside of the work of Ritchie, Englebrecht and Booysen. However, some research such as that of Wang et al. [168] suggests that shortest path search algorithms like Dijkstra and A* can be adapted to dynamic systems with time-dependant graphs.

Work in this area could make help to make the A* algorithm more robust to unexpected water draw events.

The algorithms as presented in this thesis also suffer from the obvious weakness of requiring perfect foreknowledge of water consumption patterns that will occur on a given day.

This limitation could be overcome through the development of an algorithm able to use patterns of historic behaviour to predict future demand.

One such approach, which would also incorporate knowledge of past habits of consumption with the ability to make decisions based on multiple external signals and take action to heat or not heat water is RL.

An RL approach would share many of the attributes of the shortest path search algorithms already investigated but would improve on them by learning the optimal behaviour or actions to take in many varied scenarios. The learning could include versions of the hot water cylinder models discussed in chapters 5 and 6, operate purely on feedback signals from the external grid such as pricing data and weather forecast information.

Deep reinforcement algorithms could solve many of the issues associated with the optimal heating of hot water systems, but work would also be needed to determine whether an RL

based controller would be feasible on a small, commercially available embedded micro-computer or embedded system.

10.4 Improvements to the Community Simulation Model

The community microgrid model created in MATLAB/Simulink using a bottom-up approach including historic household data. There were several shortcomings of this model that could be improved. Future work in this area could include complete and reliable household data. The data should also be taken from more up residences using modern appliances and other modern features such as LED lighting.

Some of the limitations of MATLAB/Simulink used for modelling in this way detracted from its usefulness. Simulink for example does not feature libraries with prebuilt components that can be reliably model large scale grid infrastructure and equipment.

Custom models of this kind could possibly be developed, but this would require access to very specific details and parameters that are not easily known or can be validated without particular specialist knowledge. Other open-source alternatives exist however that contain libraries of validated models that could be used to accurately simulate for example, the effect of multiple houses in a neighbourhood on equipment in the low voltage network such as transformers. An example of this would be the Pandas Power package.

Future work could focus in a more detailed package such as this that is better adapted to simulating electrical networks, and could offer better insights into the effect of operating microgrids on local low voltage electricity networks.

References

[1] H. Templeton, I. Templeton, and J. Easby, Eds., *Speeches that shaped New Zealand: 1814-1956*. Cambridge, New Zealand: Hurricane Press, 2014.

[2] "Opportunities and challenges to the future security and resilience of the New Zealand power system," Transpower, Mar. 2022.

[3] "A Roadmap For Electrification: Decarbonising transport and process heat," Transpower.

[4] New Zealand and Interim Climate Change Committee, *Accelerated electrification: evidence, analysis and recommendations, 30 April 2019*. 2019.

[5] "Power blackout probe puts onus on Electricity Authority, Transpower to do better," RNZ. Accessed: Nov. 25, 2021. [Online]. Available: https://www.rnz.co.nz/news/national/456541/power-blackout-probe-puts-onus-on-electricity-authority-transpower-to-do-better

[6] "Winter electricity blackout: Contact, Genesis actions 'generally appropriate', Authority finds," RNZ. Accessed: Dec. 16, 2021. [Online]. Available: https://www.rnz.co.nz/news/business/458034/winter-electricity-blackout-contact-genesis-actions-generally-appropriate-authority-finds

[7] "New Zealanders asked to reduce power use tomorrow morning | Transpower." Accessed: Jul. 01, 2024. [Online]. Available: https://www.transpower.co.nz/news/new-zealanders-asked-reduce-power-use-tomorrow-morning

[8] "Whakamana i Te Mauri Hiko - Empowering our Energy Future. | Transpower." Accessed: Oct. 06, 2020. [Online]. Available: https://www.transpower.co.nz/resources/whakamana-i-te-mauri-hiko-empowering-our-energy-future

[9] C. W. Gellings, "The concept of demand-side management for electric utilities," *Proceedings of the IEEE*, vol. 73, no. 10, pp. 1468–1470, Oct. 1985, doi: 10.1109/PROC.1985.13318.

[10] "Demand response guiding regulatory principles," Electricity Authority Te Mana Hiko, Aug. 2015.

[11] G. Strbac *et al.*, "Smart New Zealand Energy Futures: A Feasibility Study," Imperial College London, Jan. 2012.

[12] P. Palensky and D. Dietrich, "Demand Side Management: Demand Response, Intelligent Energy Systems, and Smart Loads," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 3, pp. 381–388, Aug. 2011, doi: 10.1109/TII.2011.2158841.

[13] IPAG, "Advice on creating equal access to electricity networks," Apr. 2019. Accessed: Jul. 26, 2021. [Online]. Available: https://www.ea.govt.nz/development/advisory-technical-groups/ipag/final-advice/

[14] EECA, "Ripple Control of Hot Water in New Zealand." EECA, New Zealand Government, Sep. 2020.

[15] Power Systems Consultants New Zealand, "Ripple Control of Hot Water in New Zealand," EECA, 2020.

[16] R. Tulabing *et al.*, "Mitigation of Local Grid Congestion Due to Electric Vehicles Through Localized Demand Control," in *2018 IEEE Innovative Smart Grid Technologies - Asia (ISGT Asia)*, May 2018, pp. 254–259. doi: 10.1109/ISGT-Asia.2018.8467768.

[17] "Winter Capacity Margin - Potential Effect of Possible Changes to Transmission Pricing," concept, Feb. 2020.

[18] Reeve, David, Comendant, Corina, and T. Stevenson, “Distributed Energy Resources - Understanding the potential | Transpower (Sapere).” Accessed: May 15, 2021. [Online]. Available: https://www.transpower.co.nz/resources/distributed-energy-resources-der-report
[19] “MATLAB.” Accessed: Jul. 07, 2024. [Online]. Available: https://au.mathworks.com/products/matlab.html
[20] A. G. Williamson, S. J. Clark, New Zealand, Energy Efficiency and Conservation Authority, and Centre for Advanced Engineering, *Domestic hot water: options and solutions*. Christchurch, N.Z.: Centre for Advanced Engineering, University of Canterbury, 2001.
[21] I. Khan, M. W. Jack, and J. Stephenson, “Identifying residential daily electricity-use profiles through time-segmented regression analysis,” *Energy and Buildings*, vol. 194, pp. 232–246, Jul. 2019, doi: 10.1016/j.enbuild.2019.04.026.
[22] J. Wright, “Evaluating solar water heating: Sun, renewable energy and climate change,” Parliamentary Commissioner for the Environment, Jul. 2012.
[23] “Relative Progress of Smart Grid Development in New Zealand,” New Zealand Smart Grid Forum, Aug. 2016.
[24] D. S. Callaway and I. A. Hiskens, “Achieving Controllability of Electric Loads,” *Proceedings of the IEEE*, vol. 99, no. 1, pp. 184–199, Jan. 2011, doi: 10.1109/JPROC.2010.2081652.
[25] V. Kapsalis and L. Hadellis, “Optimal operation scheduling of electric water heaters under dynamic pricing,” *Sustainable Cities and Society*, vol. 31, pp. 109–121, May 2017, doi: 10.1016/j.scs.2017.02.013.
[26] J. Brence, Z. Gosar, V. Serazin, J. Zupancic, and M. Gams, “Multiobjective Optimisation of Water Heater Scheduling”.
[27] A. Croft, J. Boys, G. Covic, and A. Downward, “Benchmarking optimal utilisation of residential distributed generation with load control,” in *2013 International Conference on Renewable Energy Research and Applications (ICRERA)*, Oct. 2013, pp. 821–826. doi: 10.1109/ICRERA.2013.6749866.
[28] M. Wu, Y.-Q. Bao, J. Zhang, and T. Ji, “Multi-objective optimization for electric water heater using mixed integer linear programming,” *Journal of Modern Power Systems and Clean Energy*, vol. 7, no. 5, pp. 1256–1266, 2019.
[29] M. Heleno *et al.*, “Optimizing PV self-consumption through electric water heater modeling and scheduling,” in *2015 IEEE Eindhoven PowerTech*, Jun. 2015, pp. 1–6. doi: 10.1109/PTC.2015.7232636.
[30] M. A. Z. Alvarez, K. Agbossou, A. Cardenas, S. Kelouwani, and L. Boulon, “Demand Response Strategy Applied to Residential Electric Water Heaters Using Dynamic Programming and K-Means Clustering,” *IEEE Transactions on Sustainable Energy*, vol. 11, no. 1, pp. 524–533, Jan. 2020, doi: 10.1109/TSTE.2019.2897288.
[31] M. J. Booysen, J. A. A. Engelbrecht, M. J. Ritchie, M. Apperley, and A. H. Cloete, “How much energy can optimal control of domestic water heating save?,” *Energy for Sustainable Development*, vol. 51, pp. 73–85, Aug. 2019, doi: 10.1016/j.esd.2019.05.004.
[32] R. P. van Leeuwen, J. Fink, and G. J. M. Smit, “Central model predictive control of a group of domestic heat pumps case study for a small district,” in *2015 International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)*, May 2015, pp. 1–12.

[33] M. Liu, Y. Shi, and X. Liu, “Distributed MPC of Aggregated Heterogeneous Thermostatically Controlled Loads in Smart Grid,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 2, pp. 1120–1129, Feb. 2016, doi: 10.1109/TIE.2015.2492946.

[34] A. Parisio, C. Wiezorek, T. Kyntäjä, J. Elo, and K. H. Johansson, “An MPC-based Energy Management System for multiple residential microgrids,” in *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, Aug. 2015, pp. 7–14. doi: 10.1109/CoASE.2015.7294033.

[35] H. Yan, F. Zhuo, N. Lv, H. Yi, Z. Wang, and C. Liu, “Model Predictive Control Based Energy Management of a Household Microgrid,” in *2019 IEEE 10th International Symposium on Power Electronics for Distributed Generation Systems (PEDG)*, Jun. 2019, pp. 365–369. doi: 10.1109/PEDG.2019.8807657.

[36] G. Shen, Z. E. Lee, A. Amadeh, and K. M. Zhang, “A Data-Driven Electric Water Heater Scheduling and Control System,” *Energy and Buildings*, p. 110924, Mar. 2021, doi: 10.1016/j.enbuild.2021.110924.

[37] F. Sossan, A. M. Kosek, S. Martinenas, M. Marinelli, and H. Bindner, “Scheduling of domestic water heater power demand for maximizing PV self-consumption using model predictive control,” in *IEEE PES ISGT Europe 2013*, Oct. 2013, pp. 1–5. doi: 10.1109/ISGTEurope.2013.6695317.

[38] E. Vrettos, K. Lai, F. Oldewurtel, and G. Andersson, “Predictive Control of buildings for Demand Response with dynamic day-ahead and real-time prices,” in *2013 European Control Conference (ECC)*, Jul. 2013, pp. 2527–2534. doi: 10.23919/ECC.2013.6669762.

[39] H. Aki, T. Wakui, and R. Yokoyama, “Development of a domestic hot water demand prediction model based on a bottom-up approach for residential energy management systems,” *Applied Thermal Engineering*, vol. 108, pp. 697–708, Sep. 2016, doi: 10.1016/j.applthermaleng.2016.07.094.

[40] Y. Hong, C. Chen, and H. Yang, “Implementation of demand response in home energy management system using immune clonal selection algorithm,” in *2015 IEEE Congress on Evolutionary Computation (CEC)*, May 2015, pp. 3377–3382. doi: 10.1109/CEC.2015.7257313.

[41] K. Elgazzar, H. Li, and L. Chang, “A centralized fuzzy controller for aggregated control of domestic water heaters,” in *2009 Canadian Conference on Electrical and Computer Engineering*, May 2009, pp. 1141–1146. doi: 10.1109/CCECE.2009.5090304.

[42] Q. Wu, L. Wang, and B. Li, “An optimized demand response strategy for electric water heaters and the associated impact on power system operational reliability,” in *2017 International Smart Cities Conference (ISC2)*, Sep. 2017, pp. 1–6. doi: 10.1109/ISC2.2017.8090815.

[43] T. Peirelinck, C. Hermans, F. Spiessens, and G. Deconinck, “Domain Randomization for Demand Response of an Electric Water Heater,” *IEEE Transactions on Smart Grid*, vol. 12, no. 2, pp. 1370–1379, 2021, doi: 10.1109/TSG.2020.3024656.

[44] F. Alfaverh, M. Denaï, and Y. Sun, “Demand Response Strategy Based on Reinforcement Learning and Fuzzy Reasoning for Home Energy Management,” *IEEE Access*, vol. 8, pp. 39310–39321, 2020, doi: 10.1109/ACCESS.2020.2974286.

[45] F. Ruelens, B. J. Claessens, S. Quaiyum, B. De Schutter, R. Babuška, and R. Belmans, “Reinforcement learning applied to an electric water heater: From theory to practice,” Delft University of Technology, Delft Center for Systems and Control, Technical Report 16–028, 2018. Accessed: Jan. 03, 2022. [Online]. Available: https://arxiv.org/abs/1512.00408

[46] M. Liu, S. Peeters, D. S. Callaway, and B. J. Claessens, “Trajectory Tracking With an Aggregation of Domestic Hot Water Heaters: Combining Model-Based and Model-Free Control in a Commercial Deployment,” *IEEE Transactions on Smart Grid*, vol. 10, no. 5, pp. 5686–5695, Sep. 2019, doi: 10.1109/TSG.2018.2890275.

[47] O. De Somer, A. Soares, K. Vanthournout, F. Spiessens, T. Kuijpers, and K. Vossen, “Using reinforcement learning for demand response of domestic hot water buffers: A real-life demonstration,” in *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, Sep. 2017, pp. 1–7. doi: 10.1109/ISGTEurope.2017.8260152.

[48] H. Kazmi, F. Mehmood, S. Lodeweyckx, and J. Driesen, “Gigawatt-hour scale savings on a budget of zero: Deep reinforcement learning based optimal control of hot water systems,” vol. 144, p. 159, 2018.

[49] F. Ruelens, B. J. Claessens, S. Vandael, S. Iacovella, P. Vingerhoets, and R. Belmans, “Demand response of a heterogeneous cluster of electric water heaters using batch reinforcement learning,” in *2014 Power Systems Computation Conference*, Aug. 2014, pp. 1–7. doi: 10.1109/PSCC.2014.7038106.

[50] P. Lissa, C. Deane, M. Schukat, F. Seri, M. Keane, and E. Barrett, “Deep reinforcement learning for home energy management system control,” *Energy and AI*, vol. 3, p. 100043, Mar. 2021, doi: 10.1016/j.egyai.2020.100043.

[51] P. Malysz, S. Sirouspour, and A. Emadi, “An Optimal Energy Storage Control Strategy for Grid-connected Microgrids,” *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1785–1796, Jul. 2014, doi: 10.1109/TSG.2014.2302396.

[52] V. Kapsalis, G. Safouri, and L. Hadellis, “Cost/comfort-oriented optimization algorithm for operation scheduling of electric water heaters under dynamic pricing,” *Journal of Cleaner Production*, vol. 198, pp. 1053–1065, Oct. 2018, doi: 10.1016/j.jclepro.2018.07.024.

[53] S. Edelkamp and S. Schrödl, *Heuristic search: theory and applications*. Amsterdam ; Boston: Morgan Kaufmann, 2012.

[54] “CTA-2045 Water Heater Demonstration Report Including A Business Case for CTA-2045 Market Transformation,” Bonneville Power Administration, United States, BPA Technology Innovation Project 336, Nov. 2018.

[55] M. J. Booysen and A. H. Cloete, “Sustainability through Intelligent Scheduling of Electric Water Heaters in a Smart Grid,” in *2016 IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech)*, Aug. 2016, pp. 848–855. doi: 10.1109/DASC-PICom-DataCom-CyberSciTec.2016.145.

[56] M. Roux, M. Apperley, and M. J. Booysen, “Comfort, peak load and energy: Centralised control of water heaters for demand-driven prioritisation,” *Energy for Sustainable Development*, vol. 44, pp. 78–86, Jun. 2018, doi: 10.1016/j.esd.2018.03.006.

[57] P. Du and N. Lu, “Appliance Commitment for Household Load Scheduling,” *IEEE Transactions on Smart Grid*, vol. 2, no. 2, pp. 411–419, 2011.

[58] Ritchie, Michael J., “Usage-Based Optimal Energy Control of Residential Water Heaters,” Stellenbosch : Stellenbosch University, South Africa, 2020.

[59] M. J. Ritchie, J. A. A. Engelbrecht, and M. J. Booysen, “A probabilistic hot water usage model and simulator for use in residential energy management,” *Energy and Buildings*, vol. 235, p. 110727, Mar. 2021, doi: 10.1016/j.enbuild.2021.110727.

[60] Isaacs, N.P. *et al.*, “Energy Use in New Zealand Households: Report on the Year 8 Analysis for the Household Energy End-use Project (HEEP).,” BRANZ, Judgeford, Porirua, Study Report 133, 2004.
[61] N. Isaacs *et al.*, “Energy Use in New Zealand Households: Report on the Year 10 Analysis for the Household Energy End-use Project (HEEP).,” BRANZ, Judgeford, New Zealand, 155, 2006.
[62] H. Wezenberg, “Energy management engineering : a predictive energy management system incorporating an adaptive neural network for the direct heating of domestic and industrial fluid mediums.,” 2000, doi: 10.26021/3408.
[63] S. Gyamfi, “Demand Response Assessment and Modelling of Peak Electricity Demand in the Residential Sector: Information and Communication Requirements,” Doctor of Philosophy (Phd), Univeristy of Canterbury, Christchurch, New Zealand, 2010.
[64] A. Lapthorn and N. Watson, “Smart Grid in a New Zealand Context,” 2013, Accessed: Apr. 11, 2021. [Online]. Available: https://ir.canterbury.ac.nz/handle/10092/9114
[65] N.-K. C. Nair and L. Zhang, “SmartGrid: Future networks for New Zealand power systems incorporating distributed generation,” *Energy Policy*, vol. 37, no. 9, pp. 3418–3427, Sep. 2009, doi: 10.1016/j.enpol.2009.03.025.
[66] A. Croft, “Automated Residential Demand-Side Management Strategies,” Thesis, ResearchSpace@Auckland, 2016. Accessed: Apr. 11, 2021. [Online]. Available: https://researchspace.auckland.ac.nz/handle/2292/29898
[67] J. R. Lee, “Localised Demand Control for Improved Grid Integration of Renewable Energy and Electric Vehicles,” p. 183.
[68] J. Stephenson, “Smart grid research in New Zealand – A review from the GREEN Grid research programme.”
[69] Dortans, Carsten, B. Anderson, M. Jack, and J. Stephenson, “Estimating the Technical Potential of Residential Demand Response in New Zealand: NZ GREEN Grid Project Technical Report,” University of Otago, Dunedin, New Zealand, Dec. 2018.
[70] M. W. Jack, K. Suomalainen, J. J. W. Dew, and D. Eyers, “A minimal simulation of the electricity demand of a domestic hot water cylinder for smart control,” *Applied Energy*, vol. 211, pp. 104–112, Feb. 2018, doi: 10.1016/j.apenergy.2017.11.044.
[71] R. J. W. P. Parker, “A Comparative Study of Electricity Demand Forecasting Models for Residential Hot Water at the Individual Household Level,” Master of Science, University of Otago, Dunedin, New Zealand, 2020.
[72] A. U. Rehman, “Load Disaggregation: Towards Energy Efficient Systems,” PhD, Auckland University of Technology, Auckland, New Zealand, 2020.
[73] J. Watson, N. Watson, D. Santos-Martin, A. Wood, S. Lemon, and A. Miller, “Impact of solar photovoltaics on the low-voltage distribution network in New Zealand,” *IET Generation Transmission & Distribution*, vol. 10, Oct. 2015, doi: 10.1049/iet-gtd.2014.1076.
[74] J. Suppers, “Impacts of new technologies on household electricity demand: From an individual household, a community, and a national perspective,” Thesis, The University of Waikato, 2018. Accessed: May 28, 2021. [Online]. Available: https://researchcommons.waikato.ac.nz/handle/10289/11923
[75] A. Croft, J. Boys, and G. Covic, “Net energy stored control for residential demand-side management,” in *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, Nov. 2013, pp. 8033–8038. doi: 10.1109/IECON.2013.6700476.

[76] R. Tulabing *et al.*, “Integration of Distributed Energy Resources and Enhancing Local Grid Load Factor using Localized Demand Control,” in *2020 IEEE/PES Transmission and Distribution Conference and Exposition (T D)*, Oct. 2020, pp. 1–5. doi: 10.1109/TD39804.2020.9299936.
[77] R. Tulabing, B. Mitchell, G. A. Covic, and J. Boys, “Localized Demand Control of Flexible Devices for Peak Load Management.” Accessed: Sep. 24, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/9857614/
[78] “A Flexibility Plan 1.0: what we need to do and how we can do it,” Flexforum, Aug. 2022. Accessed: Apr. 24, 2023. [Online]. Available: https://www.araake.co.nz/assets/Uploads/FlexForum-Flexibility-Plan-1.0-31-August-2022.pdf
[79] “FlexTalk: The Demand Flexibility Common Communication Protocols Project Final Report,” EECA, May 2024.
[80] “Triton hot water controller being piloted for demand-side management,” Energy Manager. Accessed: Jun. 15, 2022. [Online]. Available: https://www.energy-manager.ca/triton-hot-water-controller-being-piloted-for-demand-side-management-3014/
[81] “Armada Power — Scalable aggregation of distributed energy resources,” Armada Power. Accessed: Apr. 14, 2022. [Online]. Available: https://www.armadapower.com/software
[82] ARENA, “Storing excess solar from the grid using hot water systems,” Australian Renewable Energy Agency. Accessed: Sep. 02, 2022. [Online]. Available: https://arena.gov.au/news/storing-excess-solar-from-the-grid-using-hot-water-systems/
[83] “Rheem Active Hot Water Control,” Australian Renewable Energy Agency. Accessed: Oct. 14, 2022. [Online]. Available: https://arena.gov.au/projects/rheem-active-hot-water-control/
[84] Reeve, David, Stevension, Toby, and Comendant, Corina, “Cost Benefit Analysis of Distributed Energy Resources in New Zealand,” Sapere, Jul. 2021.
[85] J. Duncan, “Electric Vehicles Impacts on New Zealand’s Electricity System,” CAENZ, Dec. 2010.
[86] “Battery storage in New Zealand | Transpower.” Accessed: Aug. 10, 2021. [Online]. Available: https://www.transpower.co.nz/about-us/transmission-tomorrow/battery-storage-new-zealand
[87] “We* Report on Electric Vehicle Charging Trial,” Wellington Electricity Lines Ltd., Wellington, New Zealand, Jul. 2018.
[88] cortexo, “How much could the lack of urgency in the electricity sector cost us? | Know more with Cortexo.” Accessed: Jul. 04, 2021. [Online]. Available: https://www.cortexo.com/how-much-could-the-lack-of-urgency-in-the-electricity-sector-cost-us/
[89] “Sector State of Play: Energy,” New Zealand Infrastructure Commission (Te Waihanga), Discussion Document, 2020.
[90] M. Jack and K. Suomalainen, “Potential future changes to residential electricity load profiles – findings from the GridSpy dataset,” University of Otago, Christchurch, New Zealand, May 2018.
[91] “Characterisation of Australian apartment electrici...: Full Text Finder Results.” Accessed: May 09, 2022. [Online]. Available: https://resolver-ebscohost-

com.ezproxy.aut.ac.nz/openurl?sid=google&auinit=MB&aulast=Roberts&atitle=Characterisation+of+Australian+apartment+electricity+demand+and+its+implications+for+low-carbon+cities&id=doi%3a10.1016%2fj.energy.2019.04.222&title=Energy&volume=180&date=2019&spage=242&site=ftf-live

[92] M. Schlemminger, T. Ohrdes, E. Schneider, and M. Knoop, "Dataset on electrical single-family house and heat pump load profiles in Germany," *Sci Data*, vol. 9, no. 1, Art. no. 1, Feb. 2022, doi: 10.1038/s41597-022-01156-1.

[93] K. Shubhankar, B. Sturmberg, and M. Shaw, "A Review of Publicly Available Energy Data Sets," Australian National University, Canberra, ACT, Australia, No. 00120C, Oct. 2019. [Online]. Available: https://arena.gov.au/projects/wattwatchers-my-energy-marketplace/

[94] J. Kelly and W. Knottenbelt, "The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes," *Sci Data*, vol. 2, no. 1, Art. no. 1, Mar. 2015, doi: 10.1038/sdata.2015.7.

[95] L. Pereira, D. Costa, and M. Ribeiro, "A residential labeled dataset for smart meter data analytics," *Sci Data*, vol. 9, no. 1, Art. no. 1, Mar. 2022, doi: 10.1038/s41597-022-01252-2.

[96] "Smart-Grid Smart-City Customer Trial Data | Datasets | data.gov.au - beta." Accessed: Jun. 16, 2021. [Online]. Available: https://data.gov.au/dataset/ds-dga-4e21dea3-9b87-4610-94c7-15a8a77907ef/details

[97] "PECAN STREET," Pecan Street Inc. Accessed: May 09, 2022. [Online]. Available: https://www.pecanstreet.org/

[98] A. M. A. Haidar, K. Muttaqi, and D. Sutanto, "Smart Grid and its future perspectives in Australia," *Renewable and Sustainable Energy Reviews*, vol. 51, pp. 1375–1389, Nov. 2015, doi: 10.1016/j.rser.2015.07.040.

[99] B. Anderson *et al.*, "New Zealand GREEN Grid Household Electricity Demand Study 2014-2018." Accessed: Apr. 01, 2024. [Online]. Available: https://reshare.ukdataservice.ac.uk/853334/

[100] J. Mair, K. Suomalainen, D. M. Eyers, and M. W. Jack, "Sizing domestic batteries for load smoothing and peak shaving based on real-world demand data," *Energy and Buildings*, vol. 247, p. 111109, Sep. 2021, doi: 10.1016/j.enbuild.2021.111109.

[101] B. Anderson, "NZ GREEN Grid Household Electricty Demand Data (Part C) Upscaling Advice Report," University of Otago, Dunedin, New Zealand, Nov. 2019.

[102] B. Anderson, "NZ GREEN Grid Household Electricity Demand Data: EECA Data Processing (Part A) Report v1.0," University of Otago: Dunedin, Dunedin, New Zealand, 1, 2019.

[103] H. Wickham, "Tidy Data," *Journal of Statistical Software*, vol. 59, pp. 1–23, Sep. 2014, doi: 10.18637/jss.v059.i10.

[104] Dortans, Carsten, B. Anderson, and M. Jack, "NZ GREEN Grid Household Electricity Demand Data: EECA Data Processing (Part B) Report v2.1," Centre for Sustainability, University of Otago: Dunedin, 2019.

[105] W. J. Braun and D. J. Murdoch, *A first course in statistical programming with R*, Third edition. Cambridge, UK New York, NY Melbourne New Delhi Singapore: Cambridge University Press, 2021. doi: 10.1017/9781108993456.

[106] "R for Data Science (2e) - 5 Data tidying." Accessed: Jun. 03, 2024. [Online]. Available: https://r4ds.hadley.nz/data-tidy

[107] "CliFlo - the National Climate Database | NIWA." Accessed: Jul. 14, 2024. [Online]. Available: https://niwa.co.nz/climate-and-weather/cliflo-national-climate-database
[108] D. G. Ocampo, "Developing an energy-related Time-Use Diary for gaining insights into New Zealand household's electricity consumption," MSc, Univerity of Otago, Dunedin, New Zealand, 2015.
[109] B. Anderson *et al.*, "NZ Green Grid Household Electricity Demand Study," Centre for Sustainability, University of Otago: Dunedin, Dunedin, New Zealand, Aug. 2018. [Online]. Available: http://www.otago.ac.nz/centre-sustainability/
[110] K. Suomalainen, M. Jack, D. M. Eyers, R. Ford, and J. Stephenson, "Comparative Analysis of monitored and self-reported data on electricity use," *IEEE Explore*, 2017.
[111] K. Bandara, R. J. Hyndman, and C. Bergmeir, "MSTL: A Seasonal-Trend Decomposition Albgorithm for Time Series and Multiple Seasonal Patterns," University of Melbourne, Melbourne, Australia, Jul. 2021. Accessed: Sep. 21, 2022. [Online]. Available: arXiv:2107.122462v1
[112] M. Joseph, *Modern Time Series Forecasting with Python: Explore industry-ready time series forecasting using modern machine learning and deep learning*. Birmingham: Packt Publishing Limited, 2022.
[113] "Consultation RIS: Lighting | Energy Rating." Accessed: Jul. 07, 2024. [Online]. Available: https://www.energyrating.gov.au/industry-information/publications/consultation-ris-lighting
[114] J. Rouleau, A. P. Ramallo-González, L. Gosselin, P. Blanchet, and S. Natarajan, "A unified probabilistic model for predicting occupancy, domestic hot water use and electricity use in residential buildings," *Energy and Buildings*, vol. 202, p. 109375, Nov. 2019, doi: 10.1016/j.enbuild.2019.109375.
[115] T. Malatesta and J. K. Breadsell, "Identifying Home System of Practices for Energy Use with K-Means Clustering Techniques," *Sustainability*, vol. 14(15), no. 9017, p. 21, Jul. 2022.
[116] O. Dumont, C. Carmo, R. Dickes, E. Georges, S. Quoilin, and V. Lemort, "Hot water tanks : How to select the optimal modelling approach?," 2016.
[117] Y. P. Chandra and T. Matuska, "Stratification analysis of domestic hot water storage tanks: A comprehensive review," *Energy and Buildings*, vol. 187, pp. 110–131, Mar. 2019, doi: 10.1016/j.enbuild.2019.01.052.
[118] P. Kepplinger, G. Huber, M. Preißinger, and J. Petrasch, "State estimation of resistive domestic hot water heaters in arbitrary operation modes for demand side management," *Thermal Science and Engineering Progress*, vol. 9, pp. 94–109, Mar. 2019, doi: 10.1016/j.tsep.2018.11.003.
[119] A. Castell, M. Medrano, C. Solé, and L. F. Cabeza, "Dimensionless numbers used to characterize stratification in water tanks for discharging at low flow rates," *Renewable Energy*, vol. 35, no. 10, pp. 2192–2199, Oct. 2010, doi: 10.1016/j.renene.2010.03.020.
[120] "About," Thermann. Accessed: Apr. 07, 2024. [Online]. Available: https://www.thermann.com.au/about
[121] "Thermocouple Accuracy Table by Type and Temperature - NI." Accessed: Mar. 28, 2024. [Online]. Available: https://knowledge.ni.com/KnowledgeArticleDetails?id=kA00Z000000P8kSSAS&hx0026%3Bl=en-GR&l=en-NZ
[122] F. P. Incropera, D. P. DeWitt, T. L. Bergman, and A. S. Lavine, Eds., *Fundamentals of heat and mass transfer*, 6. ed. Hoboken, NJ: Wiley, 2007.

[123] "TC-08 User's Guide." Pico Technology Ltd., 2019 2015.
[124] B. Ltd, "Hot water storage cylinders," May 2023, Accessed: Apr. 06, 2024. [Online]. Available: http://www.level.org.nz/water/water-supply/hot-water-supply/storage-cylinders/
[125] *AS/NZS 4859.2:2018 Thermal insulation materials for buildings, Part 2: Design*, Sydney, Australia., Nov. 19, 2018.
[126] C. A. Cruickshank and S. J. Harrison, "Heat loss characteristics for a typical solar domestic hot water storage," *Energy and Buildings*, vol. 42, no. 10, pp. 1703–1710, Oct. 2010, doi: 10.1016/j.enbuild.2010.04.013.
[127] A. H. Cloete, "A domestic electric water heater application for Smart Grid.," Thesis, Stellenbosch : Stellenbosch University, 2017. Accessed: Apr. 16, 2021. [Online]. Available: https://scholar.sun.ac.za:443/handle/10019.1/100886
[128] *Electric water heaters Part 1: Energy consumption, performance and general requirements*, AS/NZS 4692.1:2005, Sydney, Australia., Oct. 06, 2015.
[129] *Electric water heaters Part 2: Minimum Energy Performance Standard (MEPS) requirements and energy labelling*, AS/NZS 4692.2:2005, Sydney, Australia., Sep. 14, 2005.
[130] "Report: Smart Water Heaters Could Pay Back $200 per Year in Grid Services." Accessed: Apr. 14, 2022. [Online]. Available: https://www.greentechmedia.com/articles/read/report-smart-water-heaters-could-pay-back-200-per-year-in-grid-services
[131] N. Isaacs, M. Camilleri, and L. French, "Hot Water Over Time - The New Zealand Experience," BRANZ, Melbourne, Australia, Conference Paper 132(2007), Sep. 2007.
[132] A. K. Goonesekera, "Performance Optimisation of Hot Water Cylinders".
[133] H. D. Young, R. A. Freedman, and A. L. Ford, *Sears and Zemansky's university physics*. Boston, Mass.: Addison-Wesley, 2012.
[134] J. K. Kuhn, G. F. von Fuchs, and A. P. Zob, "Developing and upgrading of solar system thermal energy storage simulation models. Technical progress report, March 1, 1979-February 29, 1980," Boeing Computer Services Co., Seattle, WA (USA), DOE/CS/34482-4; BCS-40311, May 1980. doi: 10.2172/6817726.
[135] Y. Bai, M. Yang, Z. Wang, X. Li, and L. Chen, "Thermal stratification in a cylindrical tank due to heat losses while in standby mode," *Solar Energy*, vol. 185, pp. 222–234, Jun. 2019, doi: 10.1016/j.solener.2018.12.063.
[136] Y. H. Zurigat, K. J. Maloney, and A. J. Ghajr, "A Comparison Study of One-Dimensional Models for Stratified Thermal Storage Tanks," *Journal of Solar Energy Engineering*, vol. 111, no. 204, pp. 204–210, Aug. 1989.
[137] Z. Lavan and J. Thompson, "Experimental study of thermally stratified hot water storage tanks," *Solar Energy*, vol. 19, no. 5, pp. 519–524, Jan. 1977, doi: 10.1016/0038-092X(77)90108-6.
[138] S. V. Pandiyan and J. Rajasekharan, "Recursive training based physics-inspired neural network for electric water heater modeling," *Energy Informatics*, vol. 5, no. 4, p. 58, Dec. 2022, doi: 10.1186/s42162-022-00233-4.
[139] A. A. Farooq, A. Afram, N. Schulz, and F. Janabi-Sharifi, "Grey-box modeling of a low pressure electric boiler for domestic hot water system," *Applied Thermal Engineering*, vol. 84, pp. 257–267, 2015, doi: 10.1016/j.applthermaleng.2015.03.050.
[140] M. Shad, A. Momeni, R. Errouissi, C. P. Diduch, M. E. Kaye, and L. Chang, "Identification and Estimation for Electric Water Heaters in Direct Load Control

Programs," *IEEE Transactions on Smart Grid*, vol. 8, no. 2, pp. 947–955, Mar. 2017, doi: 10.1109/TSG.2015.2492950.

[141] M. J. RItchie, J. a. A. Engelbrecht, and M. J. Booysen, "Impact of node count on energy-optimal control of stratified vertical water heaters in smart grid applications," Jul. 19, 2021, *engrXiv*. doi: 10.31224/osf.io/gbvc9.

[142] J. A. A. Engelbrecht, M. J. Ritchie, and M. J. Booysen, "Optimal schedule and temperature control of stratified water heaters," *Energy for Sustainable Development*, vol. 62, pp. 67–81, Jun. 2021, doi: 10.1016/j.esd.2021.03.009.

[143] D. Easdown, *A first course in linear algebra*, 3rd ed. Frenchs Forest, N.S.W.: Pearson SprintPrint, 2011.

[144] "Kronecker delta function - MATLAB kroneckerDelta - MathWorks Australia." Accessed: Aug. 04, 2024. [Online]. Available: https://au.mathworks.com/help/symbolic/sym.kroneckerdelta.html

[145] "Duffie: Solar engineering of thermal processes, photovolt... - Google Scholar." Accessed: Jul. 28, 2024. [Online]. Available: https://scholar.google.com/scholar_lookup?title=Solar%20Engineering%20of%20Thermal%20Processes&author=J.A.%20Duffie&publication_year=2013

[146] D. I. Wilson, *Numerical Methods with MATLAB for Engineers*. in 20th IFAC World Congress. Auckland, New Zealand, 2020.

[147] C. B. Moler, *Numerical computing with MATLAB*, Revised reprint. Philadelphia, Pa: Society for Industrial & Applied Mathematics, 2011.

[148] S. L. Eshkabilov, *Practical MATLAB Modeling with Simulink: Programming and Simulating Ordinary and Partial Differential Equations*. Fargo, United States of America: Apress Media, 2020.

[149] J. Rust, "Dynamic Programming," *The New Palgrave Dictionary of Economics*, pp. 1–26, 2008, doi: 10.1057/978-1-349-95121-5_1932-1.

[150] P. Brandimarte, *From shortest paths to reinforcement learning: A MATLAB-based tutorial on dynamic programming*. in EURO advanced tutorials on operational research. Cham: Springer, 2021.

[151] D. P. Bertsekas, *Reinforcement learning and optimal control*, 2nd printing (includes editorial revisions). Belmont, Massachusetts: Athena Scientific, 2019.

[152] R. Sedgewick and K. D. Wayne, *Algorithms*, 4th ed. Upper Saddle River: Addison-Wesley, 2011.

[153] J. E. Stout, M. G. Best, and V. L. Yu, "Susceptibility of members of the family Legionellaceae to thermal stress: implications for heat eradication methods in water distribution systems.," *Appl. Environ. Microbiol.*, vol. 52, no. 2, pp. 396–399, Aug. 1986.

[154] J. Wengrow, *A Common-Sense Guide to Data Structures and Algorithms.*, Second Edition. Pragmatic Publishers, 2020.

[155] M. J. Ritchie, J. A. A. Engelbrecht, and M. J. Booysen, "Practically-Achievable Energy Savings with the Optimal Control of Stratified Water Heaters with Predicted Usage," *Energies*, vol. 14, no. 7, Art. no. 7, Jan. 2021, doi: 10.3390/en14071963.

[156] M. J. Ritchie, J. A. A. Engelbrecht, and M. J. Booysen, "Centrally Adapted Optimal Control of Multiple Electric Water Heaters," *Energies*, vol. 15, no. 4, Art. no. 4, Jan. 2022, doi: 10.3390/en15041521.

[157] P. M. Armstrong, M. Uapipatanakul, I. Thompson, D. Ager, and M. McCulloch, "Thermal and sanitary performance of domestic hot water cylinders: Conflicting

requirements," *Applied Energy*, vol. 131, pp. 171–179, Oct. 2014, doi: 10.1016/j.apenergy.2014.06.021.
[158] R. Roberts, A. Brent, and J. Hinkley, "Reviewing the impacts of community energy initiatives in New Zealand," *Kōtuitui: New Zealand Journal of Social Sciences Online*, vol. 16, no. 1, pp. 45–60, Jan. 2021, doi: 10.1080/1177083X.2021.1877161.
[159] "COMMENT: Embedded networks - what to watch out for - Good Returns." Accessed: Jun. 07, 2022. [Online]. Available: https://www.goodreturns.co.nz/article/976513993/https//www.goodreturns.co.nz/article/976513993/comment-embedded-networks-what-to-watch-out-for.html
[160] "Authority response to RAG report on Review of secondary networks," Electricity Authority Te Mana Hiko, May 2017.
[161] "Review of secondary networks," Retail Advisory Group, Jan. 2017.
[162] "Guidelines for Metering, Reconciliation, and Registry Arrangements for Secondary Networks," Electricity Authority Te Mana Hiko, Apr. 2011.
[163] N. Watson, "Assessing the Impact of Electric Vehicle Chargers on a Low Voltage Distribution System," 2018, Accessed: Nov. 10, 2021. [Online]. Available: https://ir.canterbury.ac.nz/handle/10092/101069
[164] "Energy in New Zealand 2020," p. 68, 2019.
[165] C. Brandolino and P. Chappell, "Mostly Climate Summary," NIWA, Auckland, June 2015, Jul. 2015. [Online]. Available: https://niwa.co.nz/climate/summaries/monthly/climate-summary-for-june-2015
[166] "Consumer electricity supply arrangements," Concept Consulting, 2, Oct. 2021.
[167] "Whole Electricity System Costs, A Report for Vector," Frontier Economics, Mar. 2021.
[168] Y. Wang, Y. Yuan, Y. Ma, and G. Wang, "Time-Dependant Graphs: Definitions, Applications, and Algorithms," *Data Science and Engineering*, vol. 4, pp. 352–366, Jun. 2019.

Appendices

Appendix A : Data analysis code example

Representative MATLAB code for data analysis of GREEN grid data

Import data from text file - House 1

Script for importing data from the following text file - data for the year between 01 April 2015 - 31 March 201 (House 1) for appending New Plymouth weather data. These are the dates for which weather data has been obtained:

```
  filename: D:\AUT2021\Project 2021\GreenGridData\powerData -
WorkingCopy\NewPlymouth_Data\rf_1_all_1min_data.csv
```

Auto-generated by MATLAB on 12-May-2021 20:15:54

Setup the Import Options and import the data

```
opts = delimitedTextImportOptions("NumVariables", 7);
% Specify range and delimiter
opts.DataLines = [2 inf];  % dates between 00:00:00 01/04/15 - 15:54:59 21/03/1 3219202
[209514, 501104]
opts.Delimiter = ",";
% Specify column names and types
opts.VariableNames = ["hhID", "linkID", "dateTime_orig", "TZ_orig", "r_dateTime",
"circuit", "powerW"];
opts.SelectedVariableNames = ["r_dateTime", "circuit", "powerW"];
opts.VariableTypes = ["string", "string", "string", "string", "categorical", "categorical",
"double"];
% Specify file level properties
opts.ExtraColumnsRule = "ignore";
opts.EmptyLineRule = "read";
% Specify variable properties
opts = setvaropts(opts, ["hhID", "linkID", "dateTime_orig", "TZ_orig"], "WhitespaceRule",
"preserve");
opts = setvaropts(opts, ["hhID", "linkID", "dateTime_orig", "TZ_orig", "r_dateTime",
"circuit"], "EmptyFieldRule", "auto");
% Import the data
rf1AnnData = readtable("D:\AUT2021\Project 2021\GreenGridData\powerData -
WorkingCopy\FirstHouses\rf_01_all_1min_data.csv", opts);
```

Clear temporary variables

```
clear opts
```

Convert dateTime to datetime type and add to table

```
% Change type of dateTime from categorical to datetime type
dateString = convertvars(rf1AnnData,"r_dateTime",'string'); % Need to convert to string to use datetime fcn
dateTimeCell = table2cell(dateString);                          % Need to convert to string to use datetime fcn
dateTimes = dateTimeCell(:,1);                                  % Isolate datetime data in cell array for modification by datetime fcn
dateCells = cellstr(dateTimes);
modDateCells = datetime(dateCells,'InputFormat','yyyy-MM-dd''T''HH:mm:ssXXX','TimeZone','UTC');
modDateCells.TimeZone = '';
house1_data = rf1AnnData;                                   % Rename for readability
house1_data.dateTime = modDateCells
clear dateString dateTimeCell dateCells modDateCells;
```

Plot raw data for one year

Plots of each circuit for one year to be used as a diagnositc/troubleshooting tool and indication/visualisation of raw household data and characteristics.

```
houseVar = 1;
rawIncomer = house1_data(house1_data.circuit == 'Mains$1634',:);
rawHotWater = house1_data(house1_data.circuit == 'Hot water$1636',:);
rawKitchenAppliances = house1_data(house1_data.circuit == 'Kitchen power$1632',:);
rawLighting = house1_data(house1_data.circuit == 'Lights$1635',:);
rawOven = house1_data(house1_data.circuit == 'Range$1637',:);
rawHeatPump = house1_data(house1_data.circuit == 'Heating$1633',:);


subplot(6,1,1)
plot(rawIncomer.dateTime,rawIncomer.powerW)
ylabel('Mains')
title(['Raw Annual Data - House ',num2str(houseVar)])
subplot(6,1,2)
plot(rawHotWater.dateTime,rawHotWater.powerW)
ylabel('Hot water')
subplot(6,1,3)
plot(rawKitchenAppliances.dateTime,rawKitchenAppliances.powerW)
ylabel('Kitchen Appliance and Laundry')
subplot(6,1,4)
```

```
plot(rawLighting.dateTime,rawLighting.powerW)
ylabel('Lighting')
subplot(6,1,5)
plot(rawOven.dateTime,rawOven.powerW)
ylabel('Oven, Hob')
subplot(6,1,6)
plot(rawHeatPump.dateTime,rawHeatPump.powerW)
ylabel('Upstairs Heat Pump')
clear rawLighting rawLaundry rawKitchenAppliances rawHotWater rawIncomer rawOven
rawHeatPump rawOther;
```

```
Separate Power Variables into single table - unpacking data spreadsheet
```

%Isolate each circuit power data row and remove unnecessary columns to facilitate joining into one wide table for each datatime entry.

```
H1_Oven = house1_data(house1_data.circuit == 'Hob$2589',:);
H1_Oven = renamevars(H1_Oven,'powerW','Oven_PowerW');
H1_Oven = removevars(H1_Oven,{'r_dateTime','circuit'});
H1_HotWater = house1_data(house1_data.circuit == 'Hot Water Cpbd Heater- Cont$2586',:);
H1_HotWater = renamevars(H1_HotWater,'powerW','HW_PowerW');
H1_HotWater = removevars(H1_HotWater,{'circuit','r_dateTime'});
H1_Kitchen = house1_data(house1_data.circuit == 'Kitchen Appliances & Laundry$2588',:);
H1_Kitchen = renamevars(H1_Kitchen,'powerW','Kit_PowerW');
H1_Kitchen = removevars(H1_Kitchen,{'circuit','r_dateTime'});
H1_HeatPump = house1_data(house1_data.circuit == 'Heat Pump & Lounge$2590',:);
H1_HeatPump = removevars(H1_HeatPump,{'r_dateTime','circuit'});
H1_HeatPump = renamevars(H1_HeatPump,'powerW','HP_PowerW');
H1_HeatPump = movevars(H1_HeatPump,"dateTime","Before","HP_PowerW");
H1_Spa = house1_data(house1_data.circuit == 'Spa - Uncontrolled$2587',:);
H1_Spa = renamevars(H1_Spa,'powerW','Spa_PowerW')
H1_Spa = removevars(H1_Spa,{'circuit','r_dateTime'});
H1_Spa = movevars(H1_Spa,'dateTime',"Before",'Spa_PowerW');
house1_data = movevars(house1_data,"dateTime",'Before',"r_dateTime");
house1_data = removevars(house1_data,'r_dateTime')
```

Build up table with circuit Power Values in columns

```
house1_data = house1_data(house1_data.circuit == 'Incomer - Uncontrolled$2585',:);
house1_data = removevars(house1_data,'circuit');
house1_data = renamevars(house1_data,'powerW','Incomer_powerW')
```

```
% Join tables
joinedData = outerjoin(house8_data,H8_HeatPump,'Type','right','Keys',...
    {'dateTime'},'MergeKeys',true);
% Join tables
joinedData2 = outerjoin(joinedData,H8_HotWater,'Type','right','Keys',...
    {'dateTime'},'MergeKeys',true);
% Join tables
joinedData3 = outerjoin(joinedData2,H8_Kitchen,'Type','right','Keys',...
    {'dateTime'},'MergeKeys',true);
```

```
% Join tables
joinedData4 = outerjoin(joinedData3,H8_Laundry,'Type','right','Keys',...
    {'dateTime'},'MergeKeys',true);
% Join tables
joinedData5 = outerjoin(joinedData4,H8_Oven,'Type','right','Keys',...
    {'dateTime'},'MergeKeys',true);
house1_joinedData_Full = joinedData5
clear joinedData1 joinedData2 joinedData3 joinedData4 joinedData5;
```

Repair/replace any missing row values in table

```
house1_joinedData_Full = fillmissing(house1_joinedData_Full,"next");
```

Clear un-needed variables (Tidy up)

% This section won't be required when code is rewritten as a function that outputs only the processed array required for further work.
clear H1_HeatPump H1_Oven H1_HotWater H1_Laundry H1_Lighting...
H1_Kitchen rf1AnnData house1_data house1_joinedData2;

Prepare Annual Weather Data for combining into single table

Calls function to pre-process Niwa weather data from New Plymouth Aerodrome site for 00:00:00hrs 01/04/15 - 1:59:59 31/03/1

```
AnnWeather_NP = importNPweather("D:\AUT2021\Project 2021\GreenGridData\powerData -
WorkingCopy\NewPlymouth_Data\Hourly_npa_2015_Apr_01_2016_Mar_31.xlsx", "New Plymouth Aero (npa)", [3,
1716]);

clear Whaketu15176* house3*
```

Combine house and weather data into a single table

```
% Both tables are first converted to timetables:

house1_TT = table2timetable(house1_joinedData_Full);
AnnWeatherNP_TT = table2timetable(AnnWeather_NP);


% Timetables are merged - minutely data missing from weather array is replaced with NaN values.
combinedAnn_house1 =
synchronize(house1_TT,AnnWeatherNP_TT,'regular','fillwithmissing',"TimeStep",minutes(1));
% Full table must be converted back to table for 'fillmissing' function to remove NaN weather values
by interpolation.

combinedAnn_Table = timetable2table(combinedAnn_house1);
combinedAnn_house1 = fillmissing(combinedAnn_Table,"linear",'SamplePoints',combinedAnn_Table.dateTime);
clear combinedAnn_Table house1_TT AnnWeather_NP AnnWeatherNP_TT;
```

Label table with days of week and include durations

Energy use is likely to be different on weekdays and weekends. Day of week and whether day is weekday or weekend (binary category) added to each row.

```
% label days of week
combinedAnn_house1 = addDayOfWeek(combinedAnn_house1);
combinedAnn_house1 = movevars(combinedAnn_house1,'DayOfWeek','After',"dateTime");
% categorise/label weekends = 1, weekdays = 0
combinedAnn_house1 = isWeekend(combinedAnn_house1);
combinedAnn_house1 = movevars(combinedAnn_house1,'isWeekend','After',"DayOfWeek");
combinedAnn_house1 = addMonthOfYear(combinedAnn_house1);
combinedAnn_house1 = movevars(combinedAnn_house1,'MonthOfYear','After',"dateTime");
% display 'hour' of appliance activation for grouping later
combinedAnn_house1.HourActivation = hour(dateshift(combinedAnn_house1.dateTime,"start",'hour'));
combinedAnn_house1 = movevars(combinedAnn_house1,'HourActivation','After',"MonthOfYear")
```

Separate data into relevant months for analysis

```
house1_APR = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Apr',:);
house1_MAY = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'May',:);
house1_JUN = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Jun',:);
house1_JUL = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Jul',:);
house1_AUG = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Aug',:);
house1_SEP = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Sep',:);
house1_OCT = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Oct',:);
house1_NOV = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Nov',:);
house1_DEC = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Dec',:);
house1_JAN = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Jan',:);
house1_FEB = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Feb',:);
house1_MAR = combinedAnn_house1(combinedAnn_house1.MonthOfYear == 'Mar',:);
clear combinedAnn_house1 dateTimes
```

Save data in .mat format for storage in datastore

```
save('house1_Apr15_Mar1.mat','house1_APR', 'house1_MAY','house1_JUN','house1_JUL',...
house1_AUG','house1_SEP','house1_OCT","house1_NOV","house1_DEC","house1_JAN","house1_FEB","house1_MAR"
);
```

Appendix B: Therman hot water cylinder

ELECTRIC LARGE
HOT WATER SYSTEM

Thermann electric storage hot water units are an insulated storage vessel efficiently storing hot water, ready for use, when you need it. The Thermann range of electric water heaters offer solutions in eight different sizes to suit your needs.

RANGE FEATURES

- Commercial grade enamel and a thicker anode
- Easy installation, with water connections on both sides of tank
- Full flow pressure to all outlets
- Australian made
- 10-year tank warranty, including 1-year full parts and labour*

SPECIFICATIONS

Electric Tank

Measurements	80L	125L	160L	250L	315L	400L
Total Height (A)	925	1090	1315	1445	1765	1705
Total Diameter (B)	490	530	530	620	620	705
Outlet Height (C)	735	865	1095	1210	1530	1445
Inlet Height (D)	160	190	190	195	195	220
Electrical Entry (E)	85	100	100	105	105	130
Element Angle (K)	55°	55°	55°	72°	72°	72°
Storage Capacity	88	130	161	259	321	415
Hot Water Delivery	80	125	160	250	315	400
Net Weight Empty	41	51	59	72	93	115
Element Sizes (kW)	3.6	1.8, 3.6	2.4, 3.6	3.6	3.6	3.6
Relief Valve						
Pressure (kPa)	1000	1000	1000	1000	1000	1000
Max Inlet Pressure						
Without an ECV (kPa)	800	800	800	800	800	800
With an ECV (kPa)	650	650	650	650	650	650

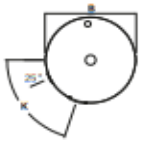
10 Cylinder

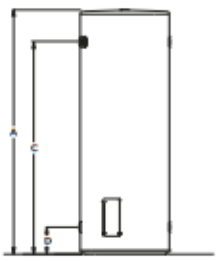
1 Parts and labour

AUSTRALIA

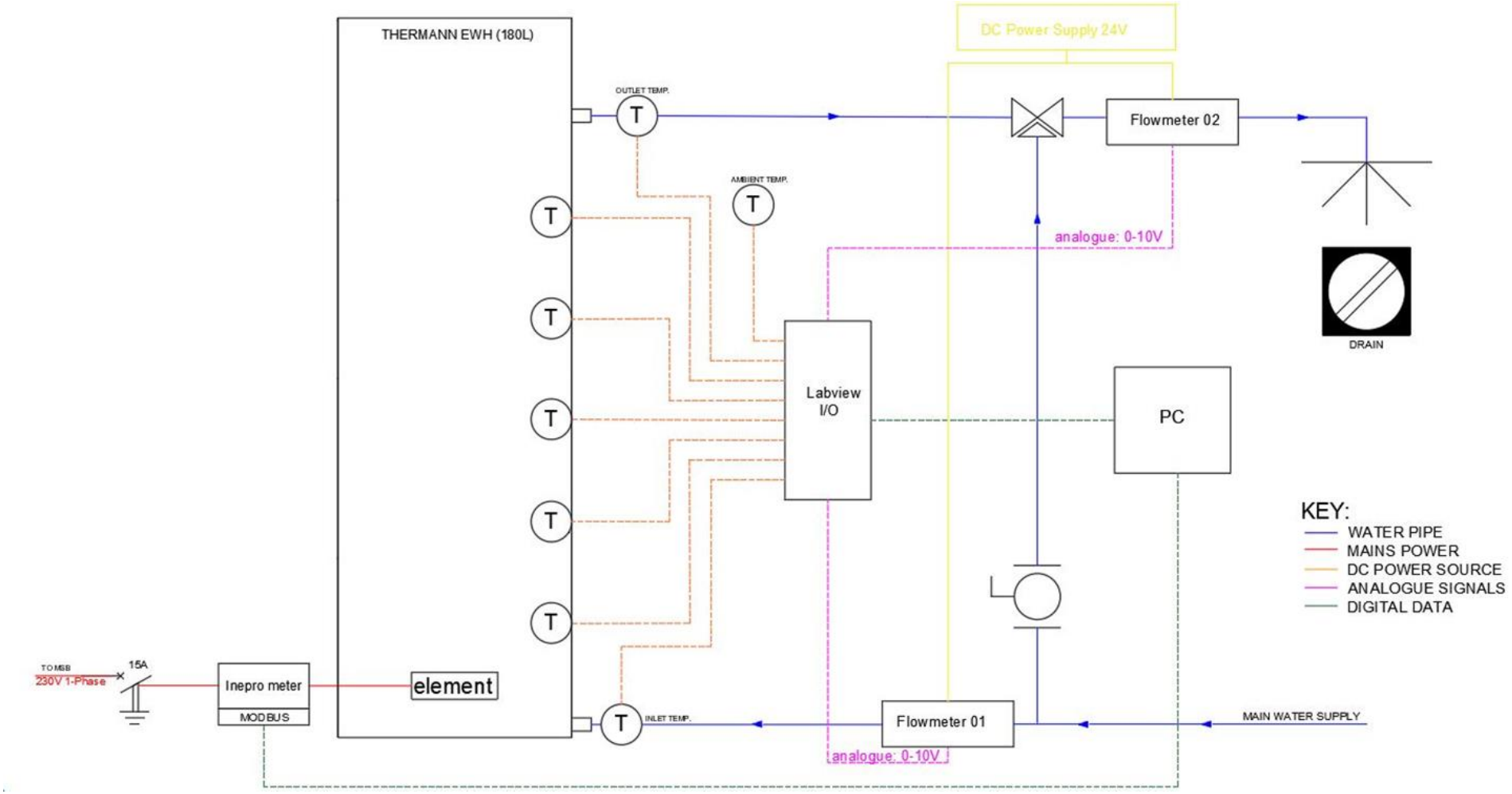
Selecting the right unit for you

	80L	125L	160L
Inlet/Outlet	Dual Handed	Dual Handed	Dual Handed
No. People (continuous)	2-3	3-4	3-5
No. People (off peak)	0	0	1-3
	250L	**315L**	**400L**
Inlet/Outlet	Dual Handed	Dual Handed	Dual Handed
No. People (continuous)	3-5	4-6	5-9
No. People (off peak)	1-3	2-4	4-6





Laboratory Hot Water Cylinder testing setup



Appendix C: Hot water cylinder testing

Hot water Cylinder testing phase: datasheets of measurement equipment

Huba Control

Flow sensor for liquid media
Type 210

The flow sensor type 210 is based on the Kármán vortex trail principle. In comparison to the OEM flow sensor (type 200), the type 210 is available with enhanced power supply and output signals and is available with and without temperature measurement.

With no moving parts the flow sensor is not sensitive to debris, has marginal pressure loss and high accuracy.

Flow range
0.5 ... 150 l/min

Nominal diameters
DN 6 / 8 / 10 / 15 / 20 / 25

Temperature measurement
-40 ... +125 °C

- Flow measuring with voltage, current, pulse or frequency output
- Temperature non-sensitive measuring principle
- Excellent media resistance (measuring element not in contact with the media)
- Wide application temperature range
- Marginal loss of pressure
- Measuring element not sensitive to debris
- Direct temperature measurement in the medium
- Drinking water approval KTW, W270, ACS, WRAS

inepro®

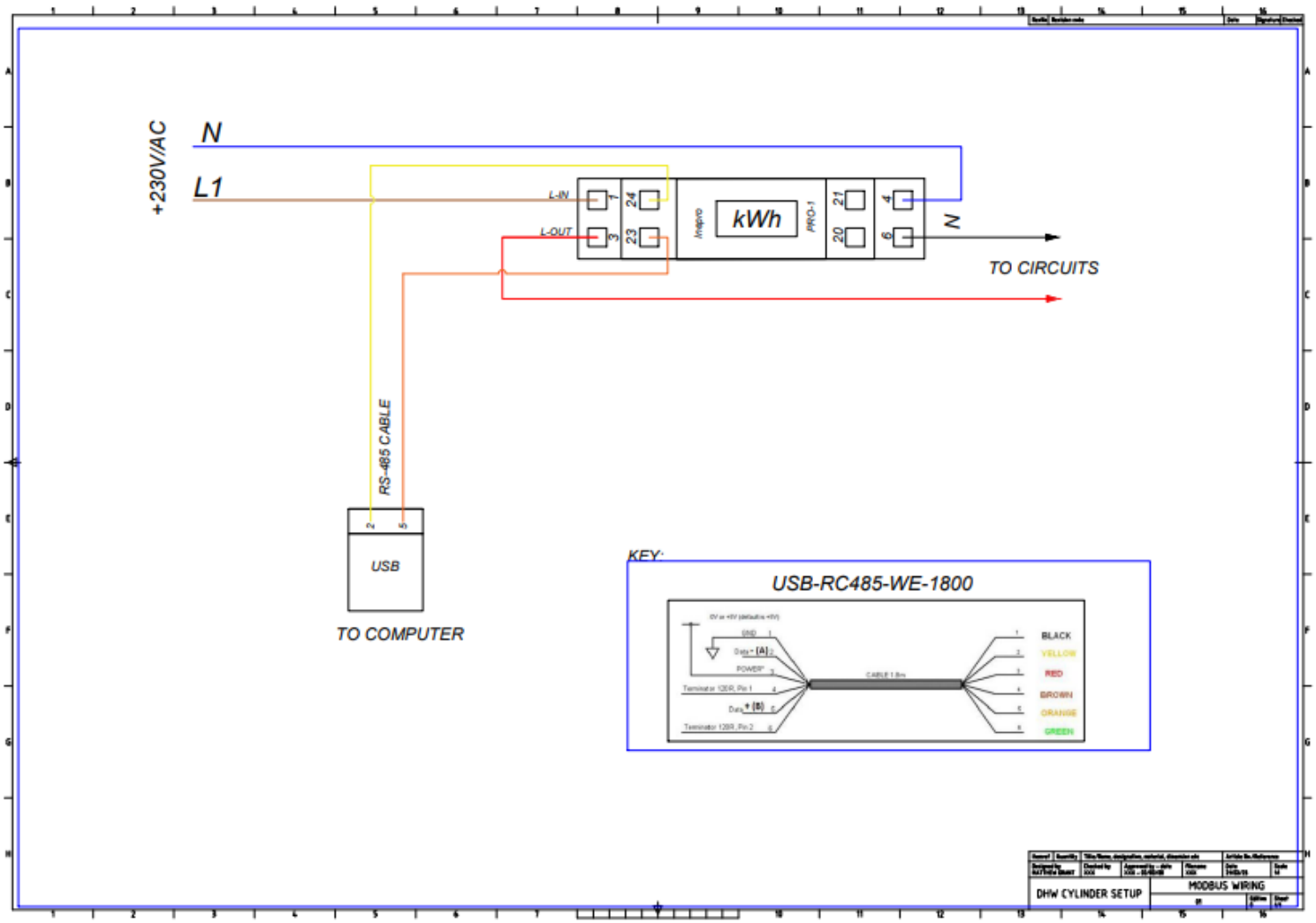
PRO1-S
PRO1-2T
PRO1-Mb
PRO1-Mod

PRO1 Series MID
Single phase energy meter

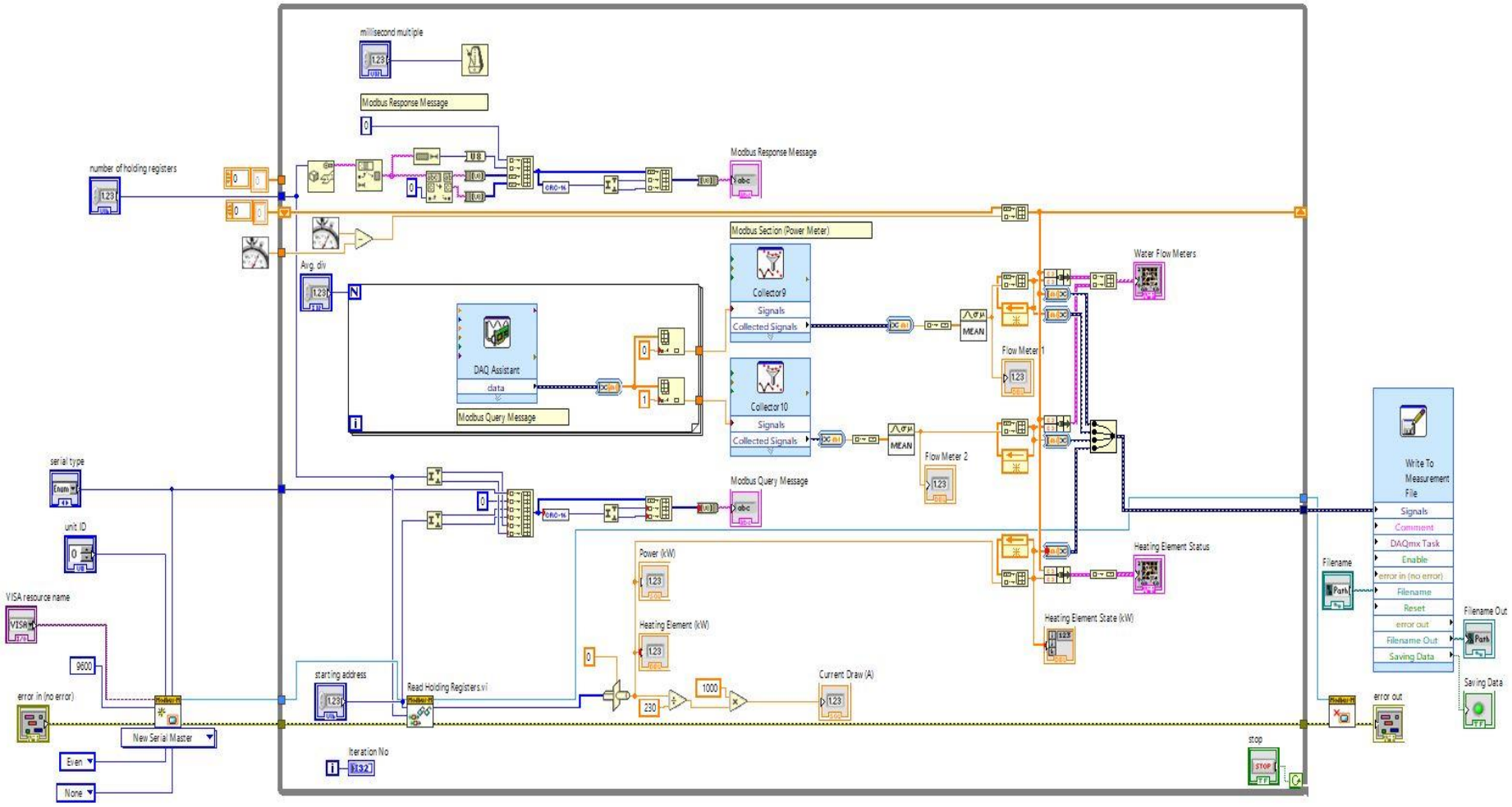
User manual

Version: 2.18-5

MODBUS connection between Inepro power meter and desktop computer:



LABVIEW code to import data from hot water cylinder testing for analysis:



Appendix D: Testing Certification

Check Testing results for 2018/2019:

Electric storage water heaters

This year, 12 models of electric storage water heaters (commonly referred to as electric hot water cylinders) were tested by an IANZ accredited laboratory (as required by the Regulations) against the relevant Minimum Energy Performance Standards (MEPS) as set out in AS/NZS 4692.2:2005. The testing results are presented in the table below. For more information about check testing please see: https://www.eeca.govt.nz/our-work/product-regulations/compliance-and-monitoring/monitoring/

Brand	Registered model number or family name	E3 Registration number	Result	Action
Superheat (300 Litre)	H300S	ZHW0162	Fail	[3]Prosecuted and fined
Atlantic (250 Litre)	886101	ZHW0220	Fail	Withdrawn from market
Atlantic (300 Litre)	896236	ZHW0221	Fail	Withdrawn from market
Thermann (176 Litre)	N180THMB124	ZHW0217	Fail	Re-registered, now compliant if TPR valve insulated
Robert Bosch (135 Litre)	7716500241	ZHW0210	Fail	Withdrawn from market
Robert Bosch (260 Litre)	7716500250	ZHW0214	Fail	Withdrawn from market
Triumph (135 Litre)	#22002 135 540 STD B.E	ZHW0111	[1]Fail	Re-registered to new Standard, now compliant
Dux Proflo (164 Litre)	160T136	AHW0228	[2]Fail	TTMRA allowance applies

All units that failed NZ MEPS, also failed to meet NZ Building Code H1/AS1 5.0.1 *Hot Water Systems* requirements.
[1]Failed NZS 4602:1988. Re-registered to AS/NZS 4692.1:2005.
[2]Complies with conditions for sale with Australian registration under the Trans-Tasman Mutual Recognition Agreement.
[3]Convicted and fined a total of $9,900 on 3 charges for breach of clause 4 and 13 of the Energy Efficiency Regulations.

Appendix E: Single Node code example

Single node (1N) solution using Euler stepping appraoch

```
tic
% Parameters
dT = 1; % Time step in seconds
t_end = 125367; % End time in seconds
num_steps = t_end / dT; % Number of time steps

% Time span for ODE solver
t_span = linspace(0, t_end, num_steps);

% Thermal tank dimensions
V = 0.176; % volume in m^3
H = 1.697; % height in m
D = 0.492; % diameter in m
r = D/2;

R_val = 1.7; % Thermal resistance
U = 1/R_val; % Heat transfer coefficient

% U constants
Atop = pi*r^2 % Total surface area of the tank in m^2
Aside = 2*pi*r*H;
SA = 2*pi*r^2 + 2*pi*r*H

us = U * 2 * pi * r * H % U value for side walls of the tank
ut = U * pi * r^2 % U value for top surface of the tank
ub = U * pi * r^2 % U value for bottom surface of the tank

Utot = [us ut ub];

Cv = 4184; % Specific heat capacity in J/(kg.K)
rho = 999.2; % Density in kg/m^3
heatingElement = 2300; % heating power (W)

% Initial conditions
T = zeros(1, length(t_span));
T(1) = 64; % initial temperature of the water in the cylinder

To = ones(size(t_span)) * 23; % ambient temperature in degrees Celsius
Tin = ones(size(t_span)) * 20; % inlet water temperature in degrees Celsius

% Flow and heater timings from experiment
Q = zeros(1, length(t_span));
F = zeros(1, length(t_span)); % Water flow vector in kg/s

F(1300:2060) = 1.47e-4; % 100L 1320:2000
F(44521:44861) = 1.47e-4; % ~50L removed
F(45346:45380) = 1.47e-4; % ~5L removed
F(45772:46232) = 1.47e-4; % ~50L removed
F(46614:46818) = 1.47e-4; % ~30L removed
F(62911:62868) = 1.47e-4; % ~30L removed
F(69291:69359) = 1.47e-4; % ~10L removed
F(74071:74207) = 1.47e-4; % ~20L removed
F(79122:79902) = 1.47e-4; % ~70L removed

Q(1494:7250) = 1;
Q(44871:45956) = 1;
Q(45347:46380) = 1;
Q(45381:46614) = 1;
Q(46728:51912) = 1;
Q(62911:65208) = 1;
Q(74124:75852) = 1;
Q(75924:76140) = 1;
Q(79740:83239) = 1;


% ode equation using Euler
    for i = 2:length(t_span)

        flowComponent = - rho*F(i)*Cv*(T(i-1);
        heatComponent = Q(i)*heatingElement;
        lossComponent = (us + ub + ut)*(T(i-1) - To(i)) / (V*rho*Cv);

        dotT = flowComponent + heatComponent + lossComponent;
        T(i) = T(i-1) + dotT*dT;
    end

toc
profile off
```

```
%% Single Node (1N) solution using ODE15s

% Time Parameters
dt = 1; % Time step in seconds
t_end = 125367; % End time in seconds
num_steps = t_end / dt; % Number of time steps

% Time span for ODE solver
t_span = linspace(0, t_end, num_steps);

% Thermal tank dimensions
V = 0.176; % volume in m^3
H = 1.697; % height in m
D = 0.492; % diameter in m
r = D/2;

R_val = 1.7; % Thermal resistance
U = 1/R_val; % Thermal conductivity (Uvalue)

% U constants
Atop = pi*r^2 % Total surface area of the tank in m^2
Aside = 2*pi*r*H;
SA = 2*pi*r^2 + 2*pi*r*H

us = U * 2 * pi * r * H % U value for side walls of the tank
ut = U * pi * r^2 % U value for top surface of the tank
ub = U * pi * r^2 % U value for bottom surface of the tank

Utot = [us ut ub];

Cv = 4184; % Specific heat capacity [J/(kg.K)]
rho = 999.2; % Density of water [kg/m^3]

% Initialize inputs
Q = zeros(1, length(t_span));
F = zeros(1, length(t_span)); % Water flow vector in kg/s

% Define flow and element states
F(1300:2060) = 1.47e-4; % 100L 1320:2000
F(44521:44861) = 1.47e-4; % ~50L removed
F(45346:45380) = 1.47e-4; % ~5L removed
F(45772:46232) = 1.47e-4; % ~50L removed
F(46614:46818) = 1.47e-4; % ~30L removed
F(62911:62868) = 1.47e-4; % ~30L removed
F(69291:69359) = 1.47e-4; % ~10L removed
F(74071:74207) = 1.47e-4; % ~20L removed
F(79122:79902) = 1.47e-4; % ~70L removed

Q(1494:7250) = 1;
Q(44871:45956) = 1;
Q(45347:46380) = 1;
Q(45381:46614) = 1;
Q(46728:51912) = 1;
Q(62911:65208) = 1;
Q(74124:75852) = 1;
Q(75924:76140) = 1;
Q(79740:83239) = 1;

q = heatingElement.*Q; % Heating power in W

T_out = 23; % Ambient temperature [degC]
T0 = 64; % Initial temp [degC]
Tinlet = ones(1, length(t_span)) * 16; % Inlet temperature in degrees Celsius

% Solver
[T, Y] = ode15s(@(t, T) Tcalc(t, T, Utot, Cv, rho, V, F, q, Tinlet, Tout, t_span), t_span, T0);
toc
outletTemp = Y;
```

```
function Tdot = Tcalc(t, T, Utot, Cv, rho, V, flow, Q, Tinlet, Tout, t_span)

    % Interpolation of input data for adaptive step size
    f = interp1(t_span, flow, t, 'linear', 'extrap');
    q = interp1(t_span, Q, t, 'linear', 'extrap');
    Tin = interp1(t_span, Tinlet, t, 'linear', 'extrap');

    us = Utot(1);
    ut = Utot(2);
    ub = Utot(3);

    % ode equation
    Tdot = (q - f*rho*Cv*(T - Tin) - (us + ub + ut)*(T - Tout)) / (V*rho*Cv);
end
```

Appendix F: 2-Node code example

2 Node Hot Water Cylinder model

```
% State-Machine Approach to 2Node hot water cylinder model - in this
% version the upper and lower temp calc function have been combined within
% a single function. Similarly, a dynamic calculation of 'us' is made with a
% function call from inside the TcalcStrat function.
% No explicit solver - Difference Equation solutions
profile on
tic
time = 125367; %3600 * 6 * 1; % No. of steps in simulation [s]
tArray = 1:1:time;

dT = 1; % 1 second

% Define states
 STATE_FULLY_MIXED = 1;
 STATE_STRATIFIED = 2;

    state = STATE_FULLY_MIXED; % Start in the fully mixed state

% Thermos tank dimensions
V = 0.175; % volume in m^3
H = 1.697; % [m]
D = 0.48; % [m]
r = D / 2;
elementValue = 2300;

% State parameters
Cv = 4184; % [J/(kg.K)] (at constant volume) [m^2/(K.s^2)]
rho = 999.2; % [kg/m^3]
R_val = 1.7;
R_valBottom = 1.3;

% DHW cylinder dimensions
SA = 2*pi*r*H + 2*pi*r^2; % [m^2] - total surface area
U_SA = (1/R_val)*SA

ut = (1/R_val)*(pi*r^2) % top UA value
ub = 0.9;%(1/R_valBottom)*(pi*r^2) %bottom has its own Uvalue
us = (1/R_val)*(2*pi*r*H)
uth = 0.6*pi*r^2; % conductivity of water across thermocline

U = [ut ub uth us];  % These are constants - us to be calculated dynamically with function call

% Storage arrays for flow/heat states
Q = zeros(1, time);
F = zeros(1, time);

% Inlet pipe water temperature array (constant for simplification)
Tinlet = ones(1, time) * 12;

% Arrays to hold output for each layer
outletTempTopBotDe = zeros(1, time); % Bottom layer
outletTempTopDe = zeros(1, time); % Top layer
qArray = zeros(1,time);
zVal = ones(1,time);
stateArray = zeros(1,time);

% Initial temperatures for both layers
T0_bot = 61.65; % Initial temp for bottom layer (from observed value)
T0_top = 64.8; % Initial temp for top layer (from observed value)
T0 = T0_top;

z = 0.6;    % Initial height of thermocline (estimated)

Tout = 23; % constant room temp

% Simulation step-size (1 sec)
tspan = [0 1];

 state = STATE_STRATIFIED;
```

```
for i = 1:length(tArray)-1

    % Current flow conditions
    f = -1*F(i)/V;
    q = Q(i)*elementValue;      % Heating element nominal value [W]
    qArray(i) = q;              % for keeping track of changes to q

    %Thermocline behaviour
    [zNewVal,resetFlagValue] = calculateThermoclineHeight(z,f);
    zVal(i) = zNewVal;  % for tracking thermocline height later
    resetFlag = resetFlagValue;

    % reset value of z if thermocline at top or bottom of cylinder
    if z < 0
        z = 0.001;

    elseif z > 1
        z = 0.999;

    end

    % State machine section
    if F(i) > 0 || (T0_bot < T0_top)   % Flow into cylinder will create new thermocline

       z = zNewVal;        %update new value of cylinder height (decimal between 0=top - 1=bottom)

        state = STATE_STRATIFIED;
        stateArray(i) = state;

    elseif     T0_bot >= T0_top

        % Turnover of hot water layers due to thermocline instability
        z = 0.99;

        state = STATE_FULLY_MIXED;
        stateArray(i) = state;

    else
        % Switch back to fully mixed state
        z = 0.99;  % Turnover of hot water layers due to thermocline instability
        state = STATE_FULLY_MIXED;
        stateArray(i) = state;

    end

    if state == STATE_FULLY_MIXED

       Tmix = TcalcFullyMixed(dT,T0,U,Cv, rho, V, f, q, Tinlet(i), Tout);

        outletTempTopDe(i) = Tmix;
        outletTempTopBotDe(i) = Tmix;

        T0_bot = Tmix; % Update initial temperature for the next time step
        T0_top = Tmix;
        T0 = Tmix;


    elseif state == STATE_STRATIFIED

        % Bottom layer calcs
        [Tbot, Ttop] = TcalcStrat(dT,T0_bot,T0_top,U, Cv, rho, V, f, q, Tinlet(i), Tout,z,H,r,R_val,resetFlag);

        T0_bot = Tbot;
        outletTempTopBotDe(i) = Tbot;

        T0_top = Ttop;
        outletTempTopDe(i) = Ttop;


    end



end
toc
profile_diffEqSolution = profile('info')
profile off
```

```
%%2 Node Hot Water Cylinder model
% State-Machine Approach to 2Node hot water cylinder model - in this
% version the upper and lower temp calc function have been combined within
% a single function. Similarly, a dynamic calculation of 'us' is made with a
% function call from inside the TcalcStrat function.
% Solver = ode15s
profile on;
tic;

time = 125367; %3600 * 6 * 1; % No. of steps in simulation [s]
tArray = 1:1:time;

timeStep = 1; % 1 second

% Define states
 STATE_FULLY_MIXED = 1;
 STATE_STRATIFIED = 2;

    state = STATE_FULLY_MIXED; % Start in the fully mixed state

% Thermos tank dimensions
V = 0.175; % volume in m^3
H = 1.697; % [m]
D = 0.48; % [m]
r = D / 2;
elementValue = 2300;

% State parameters
Cv = 4184; % [J/(kg.K)] (at constant volume) [m^2/(K.s^2)]
rho = 999.2; % [kg/m^3]
R_val = 1.7;
R_valBottom = 1.3;

% DHW cylinder dimensions
SA = 2*pi*r*H + 2*pi*r^2; % [m^2] - total surface area
U_SA = (1/R_val)*SA

ut = (1/R_val)*(pi*r^2) % top UA value
ub = (1/R_valBottom)*(pi*r^2) %bottom has its own Uvalue
us = (1/R_val)*(2*pi*r*H)
uth = 0.6*pi*r^2; % thermocline

U = [ut ub uth us];  % These are constants - us to be calculated dynamically with function call

% Actual element=ON heating times from test results of multi-day usage test
Q = zeros(1,time);
F = zeros(1, time);

% Inlet pipe water temperature array (constant for simplification)
Tinlet = ones(1, time) * 14;

% Arrays to hold output for each layer
outletTempBotode15s = zeros(1, time); % Bottom layer
outletTempTopode15s = zeros(1, time); % Top layer
qArray = zeros(1,time);
zVal = ones(1,time);
stateArray = zeros(1,time);

% Initial temperatures for both layers
T0_bot = 61.65; % Initial temp for bottom layer (from observed value)
T0_top = 63.78; % Initial temp for top layer (from observed value)
T0 = T0_top;

z = 0.95;    % Initial height of thermocline (estimated)

Tout = 23; % constant room temp

% Simulation step-size (1 sec)
tspan = [0 1];

 state = STATE_STRATIFIED;
```

```
for i = 1:length(tArray)-1

    % Current flow conditions
    f = -1*F(i)/V;
    q = Q(i)*elementValue;      % Heating element nominal value [W]
    qArray(i) = q;              % for keeping track of changes to q

    %Thermocline behaviour
    [zValReturn,resetFlagValue] = calculateThermoclineHeight(z,f);

    z = zValReturn;
    resetFlag = resetFlagValue;
    zVal(i) = zValReturn;  % for tracking thermocline height later

    % reset value of z if thermocline at top or bottom of cylinder
    if z < 0
        z = 0.001;

    elseif z > 1
        z = 0.999;

    end

    % State machine section
    if F(i) > 0 || (T0_bot < T0_top)   % Flow into cylinder will create new thermocline

       z = zValReturn;        %update new value of cylinder height (decimal between 0=top - 1=bottom)

        state = STATE_STRATIFIED;
        stateArray(i) = state;

    elseif     T0_bot >= T0_top

        % Turnover of hot water layers due to thermocline instability
        z = 0.99;

        state = STATE_FULLY_MIXED;
        stateArray(i) = state;

    else
        % Switch back to fully mixed state
        z = 0.99;  % Turnover of hot water layers due to thermocline instability
        state = STATE_FULLY_MIXED;
        stateArray(i) = state;

    end

    if state == STATE_FULLY_MIXED

       Tmixode15s = TcalcFullyMixed_ode15s(timeStep,T0,U,Cv, rho, V, f, q, Tinlet(i), Tout);

        outletTempTopode15s(i) = Tmixode15s;
        outletTempBotode15s(i) = Tmixode15s;

        T0_bot = Tmixode15s; % Update initial temperature for the next time step
        T0_top = Tmixode15s;
        T0 = Tmixode15s;


    elseif state == STATE_STRATIFIED

        % Bottom layer calcs
        [Tbot, Ttop] = TcalcStrat_ode15s(timeStep,T0_bot,T0_top,U, Cv, rho, V, f, q, Tinlet(i), Tout,z,H,r,R_val,resetFl

        T0_bot = Tbot;
        outletTempBotode15s(i) = Tbot;

        T0_top = Ttop;
        outletTempTopode15s(i) = Ttop;
        T0 = Ttop;

    end



end


% profile results saved here
profile_odeSolution = profile('info')
profile off;
toc;
```

```
function Tmixode15sed_ode15s = TcalcFullyMixed_ode15s(timestep,T0,U,Cv, rho, V, f, Q, Tinlet, Tout)\
% ode15s solution for fully-mixed state
    ut = U(1);
    ub = U(2);
    us = U(4);

    % fully mixed model
    Tmixode15sedDot = @(t,T) f*(Tinlet - T) + (Q/(Cv*rho*V)) - (1/(Cv*rho*V))*(ut + ub + us)*(T - Tout);

    tspan = [0 timestep];
    Tinit = T0;

    % solve using ode15s and return
    [t,T] = ode15s(Tmixode15sedDot, tspan, Tinit);

    Tmixode15sed_ode15s = T(end);

end


% ode15s version of 'TcalcStrat' function. Calculate the temperature of the bottom layer of the stratified tank using od
function [Tbot_ode15s, Ttop_ode15s] = TcalcStrat_ode15s(timeStep,Tbot0,Ttop0,U, Cv, rho, V, F, Q, Tinlet, Tout,z,H,r,R_v

    ut = U(1);
    ub = U(2);
    uth = U(3);
    us = U(4);

    TbotDot = @(t,T) (-1*F * (Tinlet - T) + (Q /(Cv * rho * V))*1/((1-z)) - ((ub + us) / (Cv * rho * V) * (T - Tout)*1/(
    TtopDot = @(t,T) (-1*((ut + us) / (Cv * rho * V)) * (T - Tout)*z) - (uth/(Cv*rho*V))*(Ttop0 - Tbot0);

    % time span for ode15s
    tspan = [0 timeStep];

    % initial condition for the bottom layer
    if resetFlag == 1
        TbotInit = Tinlet;
    else
        TbotInit = Tbot0;
    end

    TtopInit = Ttop0;

    % solve with ode15s
    [t,Tbot] = ode15s(TbotDot, tspan, TbotInit);
    [t,Ttop] = ode15s(TtopDot, tspan, TtopInit);

    Tbot_ode15s = Tbot(end);
    Ttop_ode15s = Ttop(end);

end

% Track thermocline height 'z'
function [zNewVal,resetFlag] = calculateThermoclineHeight(z,f)

    zNewVal_test = z + f;

        if zNewVal_test > 1
                zNewVal = 0.99;
                resetFlag = 1;

            elseif zNewVal_test < 0.01
                    zNewVal = 0.99;
                    resetFlag = 1;
            else
                zNewVal = zNewVal_test;
                resetFlag = 0;
        end

end

function [usLower,usUpper] = sideWallTherm(r,H,R_val,z)

    usUpper = ((2*pi*r*H)/R_val)*z;

    usLower = ((2*pi*r*H)/R_val)*(1-z);

end
```

Appendix G: 3-Node code example

3 Node Hot Water Cylinder model

```
% State-Machine Approach to 3-node hot water cylinder model - extending the 2-N model. In this
% version the upper and lower temp calc function have been combined within
% a single function. Similarly, a dynamic calculation of 'us' is made with a
% function call from inside the TcalcStrat function.

time = 125367; % No. of steps in simulation [s]
tArray = 1:1:time;

timeStep = 1; % 1 second
nodeNumber = 3;

% Define states
 STATE_FULLY_MIXED_1 = 1;
 STATE_STRATIFIED_2 = 2;
 STATE_STRATIFIED_3 = 3;
 STATE_STRATIFIED_4 = 4;

    %state = STATE_FULLY_MIXED; % Start in the fully mixed state

% Setup parameters
three_node_parameter_setup;

% Flow and element ON data timings obtained from laboratory whole day usage
% test
q_array_generation;
F = flowGeneration(time);

% Initial temperatures for both layers
Ttop0 = 63.78; % Initial temp for top layer (from observed value)
Tbot0 = 61.65 %55.5; % Initial temp for bottom layer (from observed value)
Tmid0 = 62.7;

T0 = Ttop0;

% Thermocline and transfer zone heights
z = 0.95;    % Initial height of thermocline (estimated)

y = z-nodeLevel_1; % extent of mid-layer below thermal transfer zone

    if y < 0        % force y positive
        y = 0;
    end

    Tout = 22; % constant room temp

% Simulation step-size (1 sec)
tspan = [0 1];


    % State machine section - these are the starting points for when
    % simulation is initialised
    % 1 = Fully mixed
    % 2 = Stratified (slight intrusion of cold water/ top two sections mixed)
    % 3 = Stratified (Colder water occupies between 1/3-2/3 of tank.
    % 4 = Stratified (Coler water now occupies >2/3 of tank, except
    % narrow layer of hot water at top of tank.
```

```
  if ((Tbot0 <= Tmid0) && (Tmid0 < Ttop0)) && (z > nodeLevel_1 && z < nodeLevel_2)  % Flow :

      state = STATE_STRATIFIED_2;


  elseif     (Tbot0 >= Tmid0 && Tmid0 < Ttop0) && z >= nodeLevel_2 && z < nodeLevel_1

      % Turnover of hot water layers due to thermocline instability

      state = STATE_STRATIFIED_3;

  elseif (Tbot0 >= Tmid0 && Tmid0 < Ttop0) && z < nodeLevel_1
      % Switch back to fully mixed state
      state = STATE_STRATIFIED_4

  elseif z < 0.99 || Tbot0 >= Ttop0

      state = STATE_FULLY_MIXED_1

  else
      state = STATE_FULLY_MIXED_1

  end

  %%%%%%%%%%%%%%%% Begin Simulation %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

r i = 1:length(tArray)-1

  % Current flow conditions
  f = 1*F(i)/V;
  q = Q(i)*elementValue;      % Heating element nominal value [W]
  qArray(i) = q;              % for keeping track of changes to q

  %Thermocline behaviour
  [zNewVal, yNewVal] = calculateThermoclineHeight(z,nodeLevel_1,f);
  zVal(i) = zNewVal;  % for tracking thermocline height later
  stateArray(i) = state;

  z = zNewVal;        %update new value of cylinder height (decimal between 0=top - 1=bottor
  y = yNewVal;

  % reset value of z if thermocline at top or bottom of cylinder
  if z < 0
      z = 0.001;      % hot water empty / all cold

  elseif z > 1
      z = 0.999;      % all hot water

  end

STATE CONDITIONS AND LOGIC
  if state == STATE_FULLY_MIXED_1

     Tmix = TcalcFullyMixed(timeStep,T0,U,Cv, rho, V, f, q, Tinlet(i), Tout);

      outletTempTop(i) = Tmix;
      outletTempMiddle(i) = Tmix;
      outletTempBottom(i) = Tmix;

      Tbot0 = Tmix; % Update initial temperature for the next time step
      Tmid0 = Tmix;
      Ttop0 = Tmix;
      T0 = Tmix;

          if F(i) > 0
              state = STATE_STRATIFIED_2;

          else
              state = STATE_FULLY_MIXED_1;
              z = 0.99;

          end
  end
```

```
if state == STATE_STRATIFIED_2

    % Bottom layer calcs
    [Tbot, Tmid, Ttop] = TcalcStrat_3N(timeStep,Tbot0,Ttop0,Tmid0,U, Cv, rho, V, f, q, Tinlet(i), Tout,z,H,r,R_val,y

    Tbot0 = Tbot;
    outletTempBottom(i) = Tbot;

    Tmid0 = Tmid;
    outletTempMiddle(i) = Tmid;

    Ttop0 = Ttop;
    outletTempTop(i) = Ttop;

            if f > 0 && z > nodeLevel_1 && z < nodeLevel_2
                state = STATE_STRATIFIED_3 ;

            elseif Tbot > Tmid

                state = STATE_STRATIFIED_4;

            else
                state = STATE_STRATIFIED_2; % stay in this state
            end

end


 if state == STATE_STRATIFIED_3

    % Bottom layer calcs
    [Tbot, Tmid, Ttop] = TcalcStrat_3N(timeStep,Tbot0,Ttop0,Tmid0,U, Cv, rho, V, f, q, Tinlet(i), Tout,z,H,r,R_val,y

    Tbot0 = Tbot;
    outletTempBottom(i) = Tbot;

    Tmid0 = Tmid;
    outletTempMiddle(i) = Tmid;

    Ttop0 = Ttop;
    outletTempTop(i) = Ttop;

            if f == 0 && Tbot0 >= Ttop0

                state = STATE_STRATIFIED_4;

            elseif f > 0 && z <= nodeLevel_1

                state = STATE_STRATIFIED_4;

            else
                state = STATE_STRATIFIED_3;

            end

    % Bottom layer calcs
    [Tbot, Tmid, Ttop] = TcalcStrat_3N(timeStep,Tbot0,Ttop0,Tmid0,U, Cv, rho, V, f, q, Tinlet(i), Tout,z,H,r,R_val,y

    Tbot0 = Tbot;
    outletTempBottom(i) = Tbot;

    Tmid0 = Tmid;
    outletTempMiddle(i) = Tmid;

    Ttop0 = Ttop;
    outletTempTop(i) = Ttop;

            if Tmid0 > Ttop0

                state = STATE_FULLY_MIXED_1;
                z = 0.99;

            end

            if f > 0 && z < nodeLevel_1

                state = STATE_STRATIFIED_4;

            end
```

```
    if state == STATE_STRATIFIED_4

        % Bottom layer calcs
        [Tbot, Ttop] = TcalcStrat_2N(timeStep,Tbot0,Ttop0,U, Cv, rho, V, f, q, Tinlet(i), Tout,z,H,r,R_val);

        Tbot0 = Tbot;
        outletTempBottom(i) = Tbot;

        outletTempMiddle(i) = Tbot;

        Ttop0 = Ttop;
        outletTempTop(i) = Ttop;
                                                % Reset thermocline to zero for tank states completely empty of comp
                if z > 0.99 || Tbot >= Ttop

                    state = STATE_FULLY_MIXED_1;
                    T0 = Ttop;

                elseif z <= 0.01
                    state = STATE_FULLY_MIXED_1;
                    T0 = Tbot;

                end

    end



end
toc
profile off
```

```
function Tmixed = TcalcFullyMixed(timestep,T0,U,Cv, rho, V, f, q, Tinlet, Tout)
    % Calculate the temperature of the tank using the Fully Mixed model

    ut = U(1);
    ub = U(2);
    us = U(4);

   flowComponent = (f*rho*(Tinlet - T0));
   heatingComponent = q/(Cv*rho*V);
   lossComponent = (1/(Cv*rho*V))*(ut + ub + us)*(T0 - Tout);

   TmixedDot =  flowComponent + heatingComponent - lossComponent;

   Tmixed = T0 + TmixedDot*timestep;

end

function [Tbot, Ttop] = TcalcStrat_2N(timeStep,Tbot0,Ttop0,U, Cv, rho, V, f, q, Tinlet, Tout,z,H,r,R_val)
% Calculate the temperature of the bottom layer of the stratified tank
    ut = U(1);
    ub = U(2);
    uth = U(3);

    if Ttop0 - Tbot0 == 0
          tClineComponent = uth/(Cv*rho*V)*0.001; % avoid divide by zero case

    else
          tClineComponent = (uth/(Cv*rho*V))*(Ttop0 - Tbot0);

    end

    [usUpper, usLower] = sideWallTherm(r,H,R_val,z);

%  Bottom half components
    lowerFlowComponent = (f * rho * (Tinlet - Tbot0)*(1-z));
    lowerHeatingComponent = (q / (Cv * rho * V))*(1/(1-z));
    lowerLossComponent = ((ub + usLower) / (Cv * rho * V)) * (Tbot0 - Tout)*(1/(1-z)) ;


    % Bottom half calculation
    TbotDot =  lowerFlowComponent + lowerHeatingComponent - lowerLossComponent + tClineComponent;
    Tbot = Tbot0 + TbotDot*timeStep;


    % Top half components
    topLossComponent = ((ut + usUpper) / (Cv * rho * V)) * (Ttop0 - Tout)*z;
    topThermoComponent = (uth*(Tbot0 - Ttop0))*z;

    % Top half calculation
    TtopDot =  -1*topLossComponent - tClineComponent;
    Ttop = Ttop0 + TtopDot*timeStep;

end
```

```
function [Tbot, Tmid, Ttop] = TcalcStrat_3N(timeStep,Tbot0,Ttop0,Tmid0,U, Cv, rho, V, f, q, Tinlet, Tout,z,H,r,R_val,y)

    ut = U(1);
    ub = U(2);
    uth = U(3);

        if Tmid0 - Tbot0 == 0
            tClineComponent_lower = uth/(Cv*rho*V)*0.001; % avoid divide by zero case

        else
            tClineComponent_lower = (uth/(Cv*rho*V))*(Tmid0 - Tbot0);

        end

    [usUpper, usMiddle, usLower] = sideWallTherm_3N(r,H,R_val,z,y);

%  Bottom section components %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    lowerFlowComponent = (f * rho * (Tinlet - Tbot0)*(1-z));
    lowerHeatingComponent = (q / (Cv * rho * V))*(1/(1-z));
    lowerLossComponent = ((ub + usLower) / (Cv * rho * V)) * (Tbot0 - Tout)*(1/(1-z)) ;

  % Bottom half calculation
    TbotDot =  lowerFlowComponent + lowerHeatingComponent - lowerLossComponent + tClineComponent_lower;
    Tbot = Tbot0 + TbotDot*timeStep;


    %  Middle section components %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    midLossComponent = (usMiddle / (Cv * rho * V)) * (Tmid0 - Tout)*y;
    midThermoTransLayer = (uth/(Cv*rho*V))*(Ttop0 - Tmid0);

    % Middle half calculation
    TmidDot =  -1*midLossComponent + midThermoTransLayer - tClineComponent_lower;    % heat transferred to bottom when b
    Tmid = Tmid0 + TmidDot*timeStep;


    % Top half components %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    topLossComponent = ((ut + usUpper) / (Cv * rho * V)) * (Ttop0 - Tout)*(z); % heat lost through top disk and side wal

    % Top half calculation
    TtopDot =  -1*topLossComponent - midThermoTransLayer;
    Ttop = Ttop0 + TtopDot*timeStep;

end



function [zNewVal,yNewVal] = calculateThermoclineHeight(z,y,f)
    % Calculate relative height [0 1] of thermocline layer and temp.
    % transfer zones in hot water cylinder.

    % NB: z==1 is bottom of tank (when fully mixed) and z==0 is top when
    % tank only contains cold water. Each unit of cold water flowing into
    % tank causes thermocline layer to retreat towards top of tank.

    % A flag not needed here, as ynewVal = 0 does not necessarily mean the
    % tank is empty.

    zNewVal = z - f;

    yNewVal = zNewVal-y;

        if yNewVal < 0
            yNewVal = 0;
        end

end

function [usLower,usUpper] = sideWallTherm(r,H,R_val,z)

    usUpper = ((2*pi*r*H)/R_val)/z;

    usLower = ((2*pi*r*H)/R_val)/(1-z);

end

function [usLower,usMiddle, usUpper] = sideWallTherm_3N(r,H,R_val,z,y)

    usUpper = ((2*pi*r*H)/R_val)*(z-y);

    usMiddle = ((2*pi*r*H)/R_val)*(y);

    usLower = ((2*pi*r*H)/R_val)*(1-z);

end
```

```
% DHW Tank parameters - sets parameters for 3-node DHW simulation

% Thermos tank dimensions
V = 0.178; % volume in m^3
H = 1.697; % [m]
D = 0.492; % [m]
r = D / 2;
elementValue = 2300;

% Reference Node Levels from top = 0
numNodes = 3;
nodeLevel_1 = 1/numNodes;
nodeLevel_2  = 2/numNodes;

% State parameters
Cv = 4184; % [J/(kg.K)] (at constant volume) [m^2/(K.s^2)]
rho = 999.2; % [kg/m^3]
R_val = 1.8;
R_valBottom = 0.4;


% DHW cylinder dimensions
SA = 2*pi*r*H + 2*pi*r^2; % [m^2] - total surface area
U_SA = (1/R_val)*SA
R = R_val / SA
%R_val = R/SA

ut = (1/R_val)*(pi*r^2); % top UA value
ub = (1/R_valBottom)*(pi*r^2); %bottom has its own Uvalue
us = (1/R_val)*(2*pi*r*H)/3;
uth = 0.6; % [d/k*Ath]

U = [ut ub uth us];  % These are constants - us to be calculated dynamically with function call

params.V = V;
params.rho = rho;
params.Cv = Cv;
params.A_cross = pi * (D/2)^2;
params.H = H;
params.U = U;
params.k = 0.6;  % Thermal conductivity of water (W/m·K)
```

Appendix H: Multi-node code examples

N-node solution – using the Euler stepping approach

```
% Euler solution - N-node model
% Simplest solution approach using single step difference equation
% Temperatures of each node are evaluated at each step

tic
% Setup
n = 10; % Number of nodes
dt = 1; % Time step
t_end = 125367; % End time
time = t_end;
num_steps = t_end / dt; % Number of time steps

t_step= 1:num_steps;

nNode_parameters; % parameters of EWH stored here

% Initialize the temperature at each node
T_series = zeros(n, num_steps);

% Initialize the temperature at each node
T0 = [61.65; 62.0; 62.3; 62.5; 62.7; 62.8; 63.1; 63.2; 63.5; 63.8];

T_series(:,1) = T0;

% thermal conductivity estimated
karray = [0.10, 0.05, 0.005, 0.005, 0.005, 0.003, 0.003,0.003, 0.001, 0.001];

% allocate space in memory for arrays
mdot = zeros(1, num_steps);
element_state = zeros(1, num_steps);

% Call to flow and heating arrays from experimental data
q_array_generation;
```

```
mdot = flowGeneration(time); % m^3/sec

% Calculate T over the time series
T_series = calculateTimeSeries(T0, k, node_positions, mdot, element_state, A_cross, A_wall, Tinlet, Tout, Cv,

toc
```

```
function [T_series, energy_series] = calculateTimeSeries(T0, k, node_positions, mdot, element_state, A_cross, ...
    A_wall, Tinlet, Tout, Cv, U, t_end, num_steps, n, mass, mass_per_node, dt)
    T_series = zeros(n, num_steps);
    T_series(:,1) = T0;
    energy_series = zeros(1, num_steps);
    energy_series(1) = sum(mass_per_node * Cv * T0);

    % node levels in tank
    h = node_positions(end) - node_positions(1);
    dz = h / n;

    % Heating element
    actual_heating_power = 2300; %[W]

    % Surface areas of tank/thermocline
    radius = sqrt(A_cross / pi);
    A_top = A_cross;
    A_bottom = A_cross;
    A_side = A_wall;

    for t_step = 2:num_steps
        current_temperatures = T_series(:, t_step-1);
        mdot_t = mdot(t_step);
        element_state_t = element_state(t_step);

        % Heat transfers across thermoclines
        conduction = zeros(n, 1);
        for i = 1:n-1
            k_eff = k * (1 + 100 * max(0, current_temperatures(i) - current_temperatures(i+1)));
            conduction(i) = k_eff * A_cross * (current_temperatures(i+1) - current_temperatures(i)) / dz;
            conduction(i+1) = conduction(i+1) - conduction(i);
        end

        convection = zeros(n, 1);
        if mdot_t > 0
            convection(1) = mdot_t * Cv * (Tinlet - current_temperatures(1));
            for i = 2:n
                convection(i) = mdot_t * Cv * (current_temperatures(i-1) - current_temperatures(i));
            end
        end

        % Heat loss from sides and top of tank
        heat_loss = zeros(n, 1);
        for i = 1:n
            if i == 1 % Bottom through floor
                heat_loss(i) = (U_bottom * A_bottom * (Tout - current_temperatures(i)) + ...
                                U_side * (A_side/n) * (Tout - current_temperatures(i))) / mass_per_node;
```

```
        % Heat loss from sides and top of tank
        heat_loss = zeros(n, 1);
        for i = 1:n
            if i == 1 % Bottom through floor
                heat_loss(i) = (U_bottom * A_bottom * (Tout - current_temperatures(i)) + ...
                                U_side * (A_side/n) * (Tout - current_temperatures(i))) / mass_per_node;
            elseif i == n % Top
                heat_loss(i) = (U_top * A_top * (Tout - current_temperatures(i)) + ...
                                U_side * (A_side/n) * (Tout - current_temperatures(i))) / mass_per_node;
            else % Middle sections
                heat_loss(i) = U_side * (A_side/n) * (Tout - current_temperatures(i)) / mass_per_node;
            end
        end

        % Calculate effective mass for heating
        effective_mass = mass_per_node;
        for i = 2:n
            if current_temperatures(i) <= current_temperatures(1) + 0.1 % add buffer to catch all
                effective_mass = effective_mass + mass_per_node;
            else
                break;
            end
        end
```

```
        % Heating calcs
        heating = zeros(n, 1);
        heating_power = element_state_t * actual_heating_power * heating_scale;


        % Diff equation of temp. change
        dotT = (conduction + convection + heat_loss + heating) / Cv;

        T_new = current_temperatures + dotT * dt;

        % Limit temp to < 100
        T_new = max(T_new, Tinlet);
        T_new = min(T_new, T_max);
        for i = 2:n
            if T_new(i-1) > T_new(i)
                T_new(i) = T_new(i-1);
            end
        end

    end
end
```

N-Node – ODE15s

```
% ode15s Solution to N-node model - uses MATLABs built in ODE15s to deal
% with system stiffness associated with on/off behaviour of heating element
% Parameters
tic
profile on;
nodes = 10; % Number of nodes
dt = 1; % Time step
t_end = 125367 % End time (36 hours in seconds)
time = 0:dt:t_end;
num_steps = length(time);

% Physical properties and dimensions
height = 1.76; % Height of the tank in meters
diameter = 0.492; % Diameter of the tank in meters
volume = 0.178; % Volume of the inner tank in cubic meters
rho = 992; % Density of water in kg/m^3
Cv = 4184; % Specific heat capacity of water in J/kg*K

%Heat transfer coefficient (increased for faster cooling)
U_top = 0.6;
U_bot = 0.9;
U_wall = 0.4;
A_cross = pi * (diameter / 2)^2; % Surface area for heat loss
A_wall = height/nodes * pi * diameter;

% Node positions (height of each node)
node_positions = linspace(0, height, nodes);
Tout = 23;
Tinlet = 16;
```

```
% Initialize the temperature at each node
T0 = [55.8; 59.9; 61; 61.3; 61.8; 62.2; 62.5; 62.7; 63.2; 63.8]; % Initial temperatures

% Thermal conductivity
k = 0.06; % estimated

% Element state and flow rate
element_state = zeros(1, num_steps);
mdot = zeros(1, num_steps);

% Call data stored from experimental test
element_state;
mdot(t_end);

% Pack parameters into a struct
params = struct('nodes', nodes, 'k', k,'A_cross', A_cross, 'A_wall', A_wall, ...
                'Tinlet', Tinlet, 'Tout', Tout, 'Cv', Cv, 'rho',rho,'U', U, ...
                'U_top', U_top, 'U_bot', U_bot,'U_wall', U_wall, 'mass_per_node', mass_per_node, 'mass',mass,...
                'heating_power', 2150, 'dz', height/nodes, 't_array', time, ...
                'mdot', mdot, 'element_state', element_state);

% Set up ODE solver
tspan = [0 t_end];
options = odeset('RelTol', 1e-6, 'AbsTol', 1e-8, MaxStep = 60);

% Initial ODE solve
[t,T] = ode15s(@(t,T) tankODE_10(t, T, params), tspan, T0, options);
toc
profile off;
```

```
% ODE function
function dotT = tankODE_10(t, T, params)
    % Unpack parameters
    nodes = params.nodes;
    k = params.k;
    U_top = params.U_top;
    U_bot = params.U_bot;
    U_wall = params.U_wall;
    A_cross = params.A_cross;
    A_wall = params.A_wall;
    Tinlet = params.Tinlet;
    Tout = params.Tout;
    Cv = params.Cv;
    rho = params.rho;
    mass_per_node = params.mass_per_node;
    heating_power = params.heating_power;
    dz = params.dz;

    % Get current mdot and element_state
    mdot_t = interp1(params.t_array, params.mdot, t, 'linear', 'extrap');
    element_state_t = interp1(params.t_array, params.element_state, t, 'nearest', 'extrap');

    % Initialize arrays
    dotT = zeros(nodes, 1);

    % Conduction
    for i = 1:nodes-1
        conduction = k * A_cross * (T(i+1) - T(i)) / dz;
        dotT(i) = dotT(i) + conduction / (mass_per_node * Cv);
        dotT(i+1) = dotT(i+1) - conduction / (mass_per_node * Cv);
    end

    % Heat loss
    for i = 1:nodes
        if i == 1
            area = A_cross + A_wall/nodes;
            heat_loss = U_bot * area * (T(i) - Tout)/ (mass_per_node * Cv);
        elseif i == nodes
            area = A_cross + A_wall/nodes;
            heat_loss = U_top * area * (T(i) - Tout)/ (mass_per_node * Cv);
        else
            area = A_wall / nodes;
            heat_loss = U_wall * area * (T(i) - Tout)/ (mass_per_node * Cv);
        end
        dotT(i) = dotT(i) - heat_loss;
    end
```

```
    % Heating
    heating_distribution = [0.2,0.1, 0.05,0.05, 0.05,0.05,0.05,0.05,...
        0.05,0.05, 0.05, 0.05, 0.05,0.05, 0.05,0.05, 0.0, 0.0, 0.0, 0.00];

    for i = 1:nodes-1

            heating = element_state_t * heating_power * heating_distribution(i);
            stratification_factor = 1 - tanh(T(i) - T(min(i+1, nodes)));
            dotT(i) = dotT(i) + (heating * stratification_factor/ (mass_per_node * Cv));

    end

    % Mass flow calcs
    if mdot_t > 0
        dotT(1) = (dotT(1) - mdot_t * rho * (T(1) - Tinlet)) / (mass_per_node * 1);
        for i = 2:nodes-1
            dotT(i) = dotT(i) - (mdot_t * rho * (T(i) - T(i-1)) / mass_per_node * 0.7);
        end

    end

    % Stratification correction
        for i = 1:nodes-1
            if T(i) > T(i+1)
                avg_temp = (T(i) + T(i+1))/2;
                dotT(i) = dotT(i) - abs((avg_temp - T(i)));
                dotT(i+1) = dotT(i+1) + abs((T(i+1) - avg_temp));
            end
        end
end
```

N-node – Analytical (Matrix)

```
% Physical properties and dimensions for N-node simulation

height = 1.76; % Height of the tank in meters
diameter = 0.492; % Diameter of the tank in meters
volume = 0.178; % Volume of the inner tank in cubic meters
rho = 992; % Density of water in kg/m^3
Cv = 4184; % Specific heat capacity of water in J/kg*K

R = 1.8; % Thermal resistance
U = 1/R; % Thermal conductivity
A_cross = pi * (diameter / 2)^2; % Surface area for heat loss
A_wall = height/n * pi * diameter; % Wall area of sections

% Node positions (height of each node)
node_positions = linspace(0, height, n);

Tout = 23;
Tinlet = 16;

% Calculate mass of water
mass = rho * volume; % Total mass of water [kg]
mass_per_node = mass/n;
```

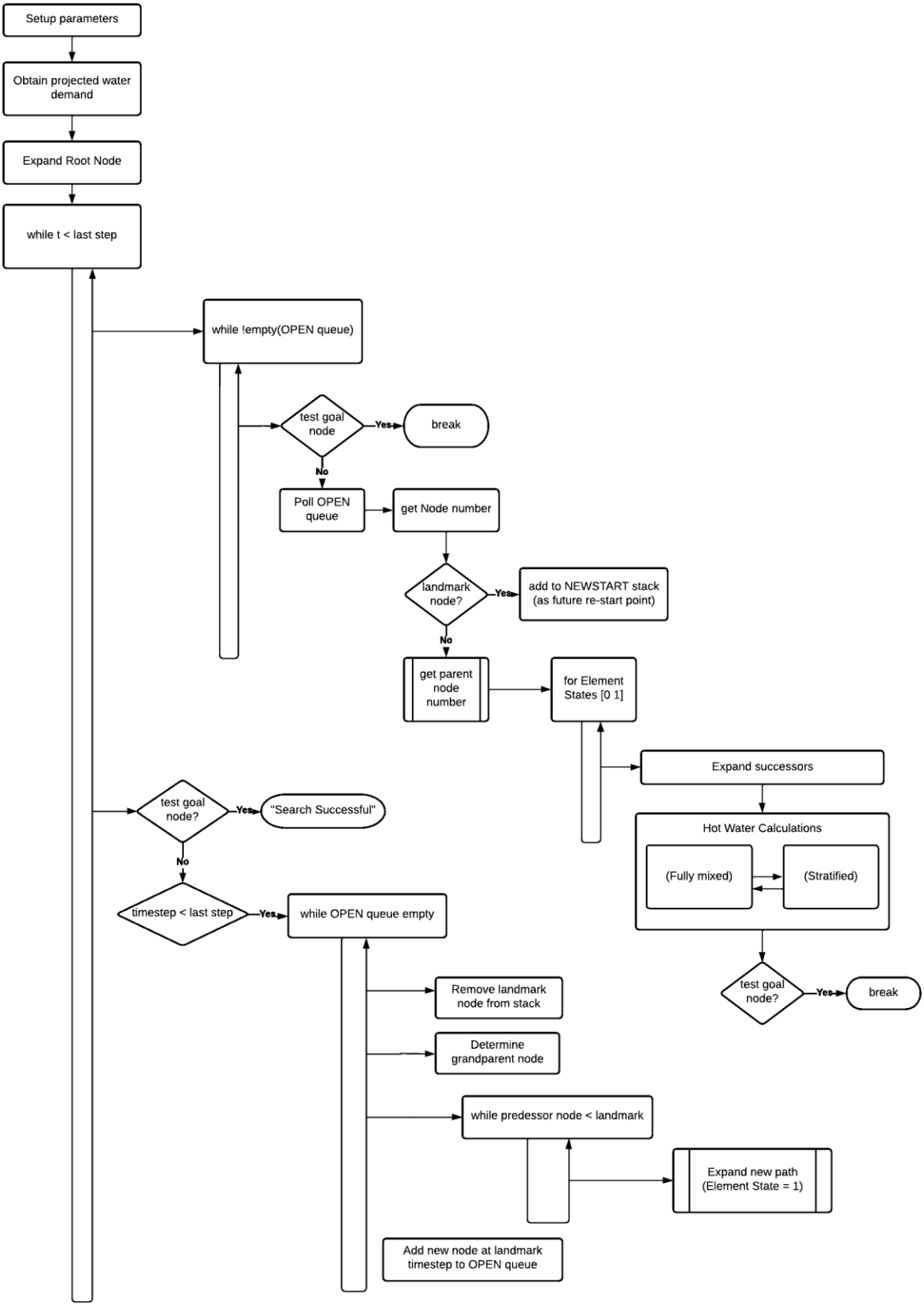
```
% Analytical solution - in this version N-node ODE's are represented as
% The main diagonal represents temperatures of nodes. Off diagonals
% represent heat loss between nodes. The b-matrix provides external inputs to
% the system including heating element and water draw/flow.

% Setup
n = 7; % Number of nodes
dt = 1; % Time step
t_end = 125367; % End time
time = t_end;
num_steps = t_end / dt; % Number of time steps

t_step= 1:num_steps;

nNode_parameters;

% Initialize the temperature at each node
T_series = zeros(n, num_steps);
% Initialize the temperature at each node
%T0 = [55.7;56.8;57.5;58.2;60.8;62.4;63.8]; % Initial temperatures (example values)
T0 = [61.65;62.0;62.7;63.1;63.2;63.5;63.8];

T_series(:,1) = T0;

% thermal conductivity
karray = [0.10, 0.05, 0.005, 0.005, 0.003, 0.003, 0.001];

% Flow rate and element ON data/timings from laboratory whole day usage
mdot = zeros(1, num_steps); % Water flow vector (kg/s)
element_state = zeros(1, num_steps); % Element state vector

q_array_generation;
F = flowGeneration(time);

% Calculate T over the time series
T_series = calculateTimeSeries(T0, karray, node_positions, mdot, element_state, A_cross, A_wall, Tinlet, Tout, ...
    Cv,rho, U, t_end, num_steps, n, mass, mass_per_node, dt);
```

```
function T_series = calculateTimeSeries(T0, karray, node_positions, mdot, element_state, A_cross, A_wall, Tinlet, Tout,

    T_series = zeros(n, num_steps); % Make space in memory
    T_series(:,1) = T0;

    % Generate mid-layer height for each node
    h = node_positions(end) - node_positions(1);
    nodeHeight = h / n;

    % Actual heating element power (W)
    elementPower = 2300;

    % Tank surface areas
    A_top = A_cross;
    A_bottom = A_cross;
    A_side = A_wall;

    % Adjust U values for different surfaces
    U_side =U;
    U_top = 1*U;
    U_bottom = 1.2*U;

    % A matrix - diagonals represent heat energy in nodes. Off-diagonals are heat exchanged between layers.
    A_const = zeros(n, n);
    for i = 1:n
        k = karray(i);
        A_const(i,i) = -(U_side*A_side/(mass_per_node*Cv) + k*A_cross/(nodeHeight^2*mass_per_node*Cv)); % diagonal eleme
        if i > 1
            A_const(i,i-1) = k*A_cross/(nodeHeight*mass_per_node*Cv);  % off-diagonal (lower)
        end
        if i < n
            A_const(i,i+1) = k*A_cross/(nodeHeight*mass_per_node*Cv); % off-diagonal (upper)
        end
    end
    A_const(1,1) = A_const(1,1) - U_bottom*A_bottom/(mass_per_node*Cv);
    A_const(n,n) = A_const(n,n) - U_top*A_top/(mass_per_node*Cv);

    for t_step = 2:num_steps
        mdot_t = mdot(t_step)*rho;
        element_state_t = element_state(t_step);
        heating_power = element_state_t * elementPower * heating_scale;

        % Update A matrix for flow
        A = A_const;
        for i = 1:n
            if i == 1
                % Bottom layer - heat exchanged upwards
                A(i,i) = A(i,i) - mdot_t/(mass_per_node);
                A(i,i+1) = A(i,i+1) + mdot_t/(mass_per_node);
            elseif i == n
                % Top layer - heat absorbed from layey below
                A(i,i) = A(i,i) - mdot_t/(mass_per_node);
                A(i,i-1) = A(i,i-1) + mdot_t/(mass_per_node);
            else
                % Mid-layers
                A(i,i) = A(i,i) - mdot_t/(mass_per_node);
                A(i,i-1) = A(i,i-1) + mdot_t/(mass_per_node);
                A(i,i+1) = A(i,i+1) + mdot_t/(mass_per_node);
            end
        end

        % Construct b vector
        b = zeros(n, 1);

        for i = 1:n
            % Heat loss to environment (remains the same for all layers)
            b(i) = b(i) + U_side*A_side*Tout / (mass_per_node*Cv);

            if i == 1
                % Bottom layer: gains energy from incoming cold water, loses to layer above
                b(i) = b(i) + mdot_t*(Tinlet - T_series(i, t_step-1)) / (mass_per_node);
                b(i) = b(i) - U_bottom*A_bottom*Tout / (mass_per_node*Cv);
            elseif i == n
                % Top layer: gains from layer below, loses energy from outgoing hot water
                b(i) = b(i) + mdot_t*(T_series(i-1, t_step-1) - T_series(i, t_step-1)) / (mass_per_node);
                b(i) = b(i) - U_top*A_top*Tout / (mass_per_node*Cv);
            else
                % Middle layers: gain from layer below, lose to layer above
                b(i) = b(i) + mdot_t*(T_series(i-1, t_step-1) - T_series(i, t_step-1)) / (mass_per_node*Cv);
            end
        end
```

```
        % Heating of bottom
        b = b + heating_power/(mass_per_node * Cv);

        % Analytic matrix solution (see Kepplinger)
        expA = expm(A * dt);
        invA = inv(A);
        invAb = invA * b;
        T_new = expA * (T_series(:,t_step-1) + invAb) - invAb;

        % Ensure temps of lower nodes stays < upper nodes
        for i = 2:n
            if T_new(i) < T_new(i-1)
                T_new(i-1:i) = mean(T_new(i-1:i));
            end
        end
        T_new = max(T_new, Tinlet);
        T_new = min(T_new, 100);  % T_max = 100°C

        T_series(:, t_step) = T_new;

    end
end
```

Appendix I: A* Search code flow diagram

Flow Diagram of A* Seach Algorithm for Optimal Hot Water control



Appendix J: A* Optimal Search code

MATLAB code of A* Seach Algorithm for Optimal Hot Water control

```
% Main Program to find optimal heating schedule for a two-node hot
% watercylinder using A* Search

% Program uses Java sources hashtable and priority queue to sort nodes and identify pathway through heating
% timeseries search space.

% In this version:
% -Extra while loop added so that backtracking steps find their way back to
%   main loop to develop new seach paths/contours if first search fails
% -Modification to allow backtracking on multiple water draw events.
% -Function to determine starting point for each backjump/backtrack start point
%   works by calculating the number of steps required to heat water needing
%   to be replaced.
% -A function subtracts the heating time from the landmark step when water draw began to estimate the time-step
%   heating should begin at.
% -The hottest node at that time step is retrieved and put back onto open
%      queue -> a new contour is created to climb over the water draw obstacle
%      and keep the water temp above the minimum comfort threshold temp.

% Further modification to allow each node to be checked for temperature and
% timestep before adding to priority queue. 2 new hashtables are added to
% achieve this.

% Functions:
%   developRootNode
%   demandStartEstimate(lastStep);
%   find();
%   length();
%   estimatedDrawCalc();
%   getParent()
%   getLandmark()
%   identifyDrawTotal();
%   newHeatingCalc();
%   maxTempNodeSearch();

tic
format shortEng;        % affects how number are displayed, NOT how computed in Matlab!

% define time step
deltaT = 120;
timeStep = 1:deltaT:86400;          % 60sec steps every minute for 24hrs = 86400steps
lastStep = 86400/deltaT; % length(timeStep);

% parameters for hot-water cylinder defined (dimensions etc.)
twoNodeEWH_params_generic;
```

```
% parameters for hot-water cylinder defined (dimensions etc.)
twoNodeEWH_params_generic;

% Initialise temperature settings;
initialTemp = 45;   % starting temp of water in tank
finalTemp = 45;
comfortTempMin = 40;
tempDiffThreshold = 0.09;

% Setup basic parameters for 2-Node simulation
simulationSetup;

backtrackFlag = 0;
landmarkCounter = 0;
lastStrucRow = 1;
constraintEasedFlag = 0;
testNodeVal = 0;
```

```
% obtain preview of estimated demand - determines number of draws, start
% time of each draw and total amount [m3] of each draw
projectedDemand = demandStartEstimate(lastStep);
demandIndex = find(projectedDemand);
numberEvents = length(demandIndex);
estimatedDraw = estimatedDrawCalc(numberEvents,projectedDemand);
```

```
% Create values of root node for beginning of simulation
developRootNode;
```

```
%%%%%%%%%%%%%%%%%%%% Main Loop %%%%%%%%%%%%%%%%%%%%

% main execution of progarm occurs in while loop, "while ~OPEN_Queue = [] "
while S(sC).timeStep < lastStep

                while (~isempty(OPEN_Queue)) %&& S(sC).timeStep < lastStep          % check priority queue to find most relevant node

                      % Test if goal node has been reached
                        if ((abs(S(sC).upperTemp - finalTemp)) <= 1 || (S(sC).upperTemp > finalTemp)) && S(sC).timeStep >= lastStep
                           OPEN_Queue.clear();
                           break;
                        end

                        if isempty(OPEN_Queue.peek)
                            sprintf('OPEN Queue empty')
                            break;
                        else
                            costKey = OPEN_Queue.poll();       % cost of highest priority node returned (lowest cost)
                        end
                                    % reset queue when we have moved forward the equivalent of 10 steps forwards
                                    if OPEN_Queue.size() > 1000

                                        OPEN_Queue.clear();

                                    end

                 % return node number (structure number)
                 sC = CLOSED_Hash.get(costKey);   % associated struct for node

                % Return values of parent needed to generate children
                 [lastUpperVol, lastLowerVol, lastEnthalpy, lastupperEnthalpy,lastlowerEnthalpy, lastUpperTemp, lastLowerTemp, ...
                 lastArrCost, lastGoCost, lastTotCost, lastzRatio, pC, stackFlag] = getParent(S,sC, stackFlag,i,projectedDemand ...
                 ,costKey,NEWSTART_Hash,newStartQueue)  ;


                    % Generate child nodes of parent for elements states [1 0]
                        for elementState = [0 1]
                                ExpandSuccessors_pruned;
                                backtrackFlag = backtrackFlag - 1;

                                        if backtrackFlag < 0
                                            backtrackFlag = 0;
                                        end
```

```
                % Test if goal node has been reached
                if (abs(S(sC).upperTemp - finalTemp) <= 1) && (S(sC).timeStep == lastStep)
                   break;
                end

        end

        % The algorithm has hit cold water draw event it cannot get over and has started to backtrack
        % (PROBLEMATIC code)

        if S(sC).timeStep > 3 && (S(sC - 2).timeStep > S(sC).timeStep) && flowEstimate2(S(sC).timeStep) > 0

            % Determine how much water will be drawn
            landmarkStep = getLandmark(demandIndex,numberEvents,S(sC).timeStep,lastStep) ;
            currentDemand = identifyDrawTotal(S(sC).timeStep,demandIndex,estimatedDraw);

            % Calculate the temperature drop this will incur and whether the comfort threshold will be breached.
            backtrackSkip = TempReductionCalc(S,sC,params,comfortTempMin,currentDemand);

            % Decide whether to backtrack or allow search to continue on current trajectory
            if backtrackSkip == 0

                OPEN_Queue.clear();     % purge priority queue to enter backtracking mode
                break;
            end
        end

end % ~isempty(OPEN_Queue)


    if S(sC).timeStep == lastStep && S(sC).upperTemp >= (finalTemp - 1)

        % Check what is left over
%
          sprintf("Search successful!")
          break;

      elseif S(sC).timeStep <= lastStep % backtrack state - a solution was not found on that contour
%
        % Enter backtracking search mode
                while OPEN_Queue.size == 0

                            lastStrucRow = sC;

                            % function to obtain landmark we have failed on
                            landmarkStep = getLandmark(demandIndex,numberEvents,S(sC).timeStep,lastStep) ;
```

```
   % function to obtain landmark we have failed on
   landmarkStep = getLandmark(demandIndex,numberEvents,S(sC).timeStep,lastStep) ;

   % calculate how much water to heat, how long it will take and what time-step to start from
   currentDemand = identifyDrawTotal(S(sC).timeStep,demandIndex,estimatedDraw);  % actual draw total that caused temp drop
   heatingTime = newHeatingCalc(deltaT,params,comfortTempMin,currentDemand);

   restartTimeStep = floor(restartTimeStepCalc(landmarkStep,heatingTime));

   % The energy required is more than the system can supply in the time available.
   % Start heating at timestep 1 anyway
   if restartTimeStep < 0
       restartTimeStep = 1;
       constraintEasedFlag = 1;
   end

   % search through all nodes in that timestep and return hottest node
   [restartCost, restartNode] = maxTempNodeSearch(CLOSED_Hash,CLOSED_Time,CLOSED_Temp,restartTimeStep);
```

```
          % forced period of heating to lift height of contour
              while S(sC).timeStep < landmarkStep

                  ExpandNewPathNodes_pruned;
                    backtrackFlag = 10;
                  lastStrucRow = sC;
              end
```

```
                                        % add this node to OPEN_Queue as next starting point of search
                                        OPEN_Queue.clear();
                                        OPEN_Queue.add(S(sC).totCost);
                                        CLOSED_Hash.put(S(sC).totCost,S(sC).structNumber);
                                        CLOSED_Time.put(S(sC).totCost,S(sC).timeStep);
                                        CLOSED_Temp.put(S(sC).totCost,S(sC).upperTemp);
                                        sprintf('New Search started')
                                        backtrackFlag = 10;


                                        break;


                                end   %while isempty(OPEN_Queue)

                        end % if ~goal state


end % ~goal state while loop
```

```
% Develop values of parent Node parameters
nodeState = 1;      % Start in fully mixed state

i = 1;
    CurrTimeStep = 1;
    ambTemp = ambTempCalc(1);      %ewhprofile0.Summer_Ambient_Temperature(i);

    % new volume of respective layers
    thisFlow = flowEstimate2(1); %ewhprofile0_summer.Summer_Water_Consumption(i)/1000;  %[m^3/min]

    lastUpperVol = upperVol;    % starts as full volume of tank

    thisUpperVol = lastUpperVol - thisFlow*deltaT/60;   % [m^3] in this timestep
    thisLowerVol = params.volume - thisUpperVol;          % [m^3]

    zRatio = zHeightCalc(upperVol,params.volume,params.height);

    %%%%%%%%%%%%% Losses will be recalculated based on change in thermocline height caused by new inflow of cold water %%%%%%%%
    %%%%%% Possible cases causing change in thermocline height shown below %%%%%%%%%%%%%%%%%%
    % This section kept in for first step because flow demand in first step possibly not zero (but most probably is zero)

    % 1. The cylinder has been emptied of hot water - the entire tank will
    % take temp value of the inlet water temperature and thermocline level will be reset
        if   zRatio <= 0.01       % 0.01 to avoid divide by zero case
             nodeState = 1;
             upperVol = params.volume;        % reset height counter and go to next timestep
             zRatio = zHeightCalc(upperVol,params.volume,params.height);
             lowerTemp = Tin;
             upperTemp = Tin;  % cold water event has occured


% 2. The two layers have equalised temperture and turnover has occurred OR the tank is full of hot water
    elseif thisFlow == 0 && ((lowerTemp >= upperTemp) || zRatio >= 1)
          nodeState = 1;
          upperVol = params.volume;        % reset height counter and go to next timestep
          zRatio = zHeightCalc(upperVol,params.volume,params.height);


% 3. Demand for water - cold water has flown into tank and caused stratification
    elseif (thisFlow > 0 || (thisFlow == 0 && (lowerTempRecord ~= upperTempRecord)))%&& all(lowerTempRecord(i) < upperTempRecord(i)))
          nodeState = 2;      % go to stratified state because cold water has entered tank and is still being heated

    end
```

```
%%%%%%%%%%%%%%%% Standing Loss calculations for step 1 ONLY %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Update relative surface areas for two nodes - this will remain constant for now
[upperSA, lowerSA] = surfAreaCalc(zRatio,params.rad,params.height);

% Represent energy lost from the tank during timestep due to thermal losses
% (Eout)
 % upper layer standing loss [w] - in mixed state z=1 so this accounts for almost all of standing loss
upperLoss = upperStandingLossCalc(params.Rval,upperSA,upperTemp,ambTemp);

% lower layer standing loss
lowerLoss = lowerStandingLossCalc(params.Rval,lowerSA,lowerTemp,ambTemp);

% thermal conductivity of thermocline at this time step - is dependant on water temp
thermConduc = ThermalConducCalc(ambTemp);
layerThickness = 1e-3;    % estimated value - (assumed to be infinitely small?) [m]

% heat exchanged across thermocline
[topLoss, botGain] = thermoclineLoss(thermConduc,layerThickness,params.rad,upperTemp,lowerTemp);

usageLoss = upperUsageCalc(params.Cv,params.rho,thisFlow,upperTemp,deltaT);       % heat leaving top of cylinder due to usage

% This should account for the energy in the cold water inserted into the lower layer.
coldFeedEnergy = coldInletCalc(params.Cv,params.rho,thisFlow,params.Tin,deltaT);


elementOutput = elementState * params.Q;



    %%%%%%%%%%%%%%%%%%%%%%%% First Node created %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Tie elements calculated into elements of first Root Node struct.
% These values don't need to be calculated as they are known already for
% the first step? (not true - what if Tinitial changes?)

% root Node elements
S(sC).structNumber = sC;
S(sC).upperVol = upperVol;
S(sC).lowerVol = lowerVol;

S(sC).Enthalpy = initialEnthalpy;
S(sC).upperEnthalpy = upperEnthalpy;
S(sC).lowerEnthalpy = lowerEnthalpy;

S(sC).lowerTemp = lowerTemp;
S(sC).upperTemp = upperTemp;

S(sC).ArrCost = 0;
S(sC).goCost = costGoHorz2(i,lastStep,finalTemp,params) + costGoVert(initialEnthalpy,finalTemp,params);
S(sC).totCost = S(sC).ArrCost + S(sC).goCost;

S(sC).zRatio = zRatio;
S(sC).elementState = elementState;
S(sC).nodeState = nodeState;
S(sC).timeStep = i;
S(sC).structParent = pC;


% send to priority queue (and hashtable?
         CLOSED_Hash.put(S(sC).totCost,S(sC).structNumber) ;       % order reversed: key = cost, value = Enthalpy
         CLOSED_Time.put(S(sC).totCost,S(sC).timeStep) ;
         CLOSED_Temp.put(S(sC).totCost,S(sC).upperTemp) ;
% only add to priority queue if node is feasible - conditional
         % logic required - if not feasible need to be able to retrieve the
         % next element in the queue.
         OPEN_Queue.add(S(sC).totCost);
```

```
function projDemandArray = demandStartEstimate(lastStep)
%PROJDEMANDARRAY Scans predicted demand profile for the day
% Builds an array to represent first time step of each hot water demand
% period. This acts as a marker in case the heating profile is not
% sufficient clear this purturbation to the heating schedule. The algorith
% will return to the parents and grandparents of this node until a new
% heating trajectory/pathway is found that can clear purturbation without
% crossing the threshold comfort temperature.

projDemandArray = zeros(1,lastStep);

    for j = 1:lastStep

        if j == 1 && flowEstimate2(1) > 0

            projDemandArray(1) = 1;

        elseif j > 1 && flowEstimate2(j) > 0 && flowEstimate2(j-1) == 0

                projDemandArray(j) = 1;
        else
                projDemandArray(j) = 0;
        end

    end
end
```

```
function estimatedDraw = estimatedDrawCalc(drawCount,projectedDemand)
%ESTIMATEDDRAW Find the begininning time step of each demand episode and
%count the number of litres consumed until the last step of the episode
%(when flow -> 0)
%   Identify each flow event start point. From this value, loop through
%   successive steps in time until the flow == 0, then move to next
%   starting point. Ouput is an array total volume values
finalArray = zeros(1,drawCount);
drawIndex = find(projectedDemand);

for j = 1:length(drawIndex)

    totalFlow = 0;
    startTime = drawIndex(j);

        while flowEstimate2(startTime)> 0

                totalFlow = totalFlow + flowEstimate2(startTime);
                startTime = startTime + 1;

        end

     finalArray(j) = totalFlow;

end

estimatedDraw = finalArray;
```

```
function [lastUpperVol, lastLowerVol, lastEnthalpy, lastupperEnthalpy,lastlowerEnthalpy, lastUpperTemp, lastLowerTemp, ...
    lastArrCost, lastGoCost, lastTotCost, lastzRatio, pC, stackFlag] = getParent(S,sC,stackFlag,i,projectedDemand,...
    costKey,NEWSTART_Hash,newStartQueue)

% GETPARENT Obtains structure node number of parent node and returns key
% values of the parent structure.

                    % Check whether this is likely to be a landmark node - a node for the algorithm to begin searching from in
                    % event of violating the comfort temperature threshold.

                    thisNodeTime = S(sC).timeStep;


                        if projectedDemand(thisNodeTime) == 1 && ~(NEWSTART_Hash.containsKey(costKey))

                            % possible landmark node to return to in event of comfort temp violation
                            % - only if this node hasn't already been on the stack (stackFlag = 0 case)

                            if stackFlag == 0

                            % this will be the first node searched for new heating trajectory/pathway
                            % -> same node keeps being readded to queue with each loop!
                            newStartQueue.add(costKey);
                            clear thisNodeTime; % tidy up for housekeeping

                             else
                                 stackFlag = 0; % reset flag
                            end
                        end


%%%%%%%%%%%%%%%% Parent node designated and values recorded for storage and use later to develop children nodes %%%%%%%%%%%%%%%
%recover parent node number
        if i == 1
                pC = 1;
        else
            pC = sC;                        % The node at sC becomes the new parent for the next step
        end

% list parent parameters where i-1 used
lastUpperVol = S(pC).upperVol;
lastLowerVol = S(pC).lowerVol;

lastEnthalpy = S(pC).Enthalpy;
lastupperEnthalpy = S(pC).upperEnthalpy;
lastlowerEnthalpy = S(pC).lowerEnthalpy;
     if isempty(lastlowerEnthalpy)
          lastlowerEnthalpy = 0;
     end

%lastMixedTemp = S(pC).newTempMixed;
lastLowerTemp = S(pC).lowerTemp;
lastUpperTemp = S(pC).upperTemp;

lastArrCost = S(pC).ArrCost;
lastGoCost = S(pC).goCost;
lastTotCost = S(pC).totCost;      % check accuracy of varialbe names
lastzRatio = S(pC).zRatio;
```

```
function landmark = getLandmark(demandIndex,eventNumber,currentTimestep,lastStep)
%GETLANDMARK Select which landmark is closest to current node
%   Take current time step of active node. Nearest landmark will be
%   first landmark index value after current time step.


for j = 1:eventNumber

    if j == 1 && currentTimestep < lastStep

         % Will always return first start index if current < second index start time
        landmark = demandIndex(1);
        %break;


    elseif j > 1 && j < eventNumber

        if currentTimestep > demandIndex(j-1) && currentTimestep <= demandIndex(j+1)

            landmark = demandIndex(j);
            %break;
        end


    elseif j == eventNumber

        if currentTimestep > demandIndex(end - 1)

            landmark = demandIndex(end);
           % break;
        end
    end
end
```

```
function thisOutput = identifyDrawTotal(currentStep,demandIndex,estimatedDraw)
%IDENTIFYDRAWSTEP Return total draw for current draw episode
%   Sort through EstimatedDraw array of total volumes drawn with each time
%   step. Find the value that corresponds to the current time step.

%drawCount = drawCountFunc(projectedDemand); % Returns number of draw events in day
%drawValues = estimatedDrawCalc(drawCount,projectedDemand); %returns array of totals for each main draw event [m^3]
%startTimes = find(projectedDemand); % array of start time indexes for each draw event
flowDemand = 0;

if length(estimatedDraw) <= 1

    flowDemand = estimatedDraw(1);

else

    for i = 1:length(estimatedDraw)-1

        if currentStep < demandIndex(1)

            flowDemand = estimatedDraw(1);

        elseif currentStep >= demandIndex(i) && currentStep < demandIndex(i+1)

           flowDemand = estimatedDraw(i);
           break;

        else
            flowDemand = estimatedDraw(end); % Catches case for last entry of drawValues (no i+1 startTime)

        end

    end
end
 thisOutput = flowDemand;

end
```

```
%EXPANDCHILDREN Calculates the children node of the pC
%   Hot water cylinder model (state machine with 2 states). Output depends
%   on whether heating element = 1 or 0.


                            % node labels generated to enable tree ordering
                            sC = lastStrucRow + 1;



                            % designating labelling parameters for new node
                            S(sC).structNumber = sC; % New node labelled with nodestructNumber
                            S(sC).structParent = pC; % pC also labelled - alternative to extra hashtable?


                            % timestep depends on timestep of pC
                            i = S(pC).timeStep + 1;

                            S(sC).timeStep = i;         % i updated in structure based on incremented value of pC node

                            if i > lastStep
                                return;
                            end
```

```
%%%%%%%%%%%% Parameters for current nodes
            ambTemp = ambTempCalc(i);      %ewhprofile0.Summer_Ambient_Temperature(i);

            % new volume of respective layers
            thisFlow = flowEstimate2(i); %ewhprofile0_summer.Summer_Water_Consumption(i)/1000;  %[m^3/min]

            thisUpperVol = lastUpperVol - thisFlow*deltaT/deltaT;   % [m^3] in this timestep
            S(sC).upperVol = thisUpperVol;

            thisLowerVol = params.volume - thisUpperVol;          % [m^3]
            S(sC).lowerVol = thisLowerVol;

            zRatio = zHeightCalc(thisUpperVol,params.volume,params.height);
            S(sC).zRatio = zRatio;

            lastArrCost = S(pC).ArrCost;
```

```
%%%%%%%%%%%% Losses will be recalculated based on change in thermocline height caused by new inflow of cold water %%%%%%%%
%%%%%% Possible cases causing change in thermocline height shown below %%%%%%%%%%%%%%%%%%%
% This sectin kept in for first step because flow demand in first step possibly not zero (but most probably is zero)

% 1. The cylinder has been emptied of hot water - the entire tank will
% take temp value of the inlet water temperature and thermocline level will be reset
    if   zRatio <= 0.01       % 0.01 to avoid divide by zero case
         thisNodeState = 1;
         thisUpperVol = params.volume;        % reset height counter and go to next timestep
         zRatio = zHeightCalc(thisUpperVol,params.volume,params.height);
         thisLowerTemp = params.Tin;
         thisUpperTemp = params.Tin;  % cold water event has occured

         % update current node
         S(sC).zRatio = zRatio;
         S(sC).nodeState = thisNodeState;
         S(sC).upperVol = thisUpperVol;

% 2. The two layers have equalised temperture and turnover has occurred OR the tank is full of hot water
    elseif thisFlow == 0 && ((lastLowerTemp >= lastUpperTemp) || zRatio >=1)
          thisNodeState = 1;
          thisUpperVol = params.volume;        % reset height counter and go to next timestep
          zRatio = zHeightCalc(thisUpperVol,params.volume,params.height);
          thisLowerVol = params.volume - thisUpperVol;
          S(sC).upperVol = thisUpperVol ;           % [m^3]
          S(sC).lowerVol = thisLowerVol;
          S(sC).lowerEnthalpy = 0;

          % update current node
          S(sC).zRatio = zRatio;
          S(sC).nodeState = thisNodeState; % IS THIS STEP NECESSARY?

% 3. Demand for water - cold water has flown into tank and caused stratification
    elseif (thisFlow > 0 || (thisFlow == 0 && (lastLowerTemp ~= lastUpperTemp)))%&& all(lowerTempRecord(i) < upperTempRecord(i)))
        zRatio = zRatioCalc(thisUpperVol,params.volume);
        thisNodeState = 2;      % go to stratified state because cold water has entered tank and is still being heated

        % update current node
         S(sC).zRatio = zRatio;
         S(sC).nodeState = thisNodeState;
    end
```

```
%%%%%%%%%%%%%%%% Standing Loss calculations %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Update relative surface areas for two nodes - this will remain constant for now
[upperSA, lowerSA] = surfAreaCalc(zRatio,params.rad,params.height);

% Represent energy lost from the tank during timestep due to thermal losses
% (Eout)
% upper layer standing loss [w] - in mixed state z=1 so this accounts for almost all of standing loss
thisUpperLoss = upperStandingLossCalc(params.Rval,upperSA,lastUpperTemp,ambTemp);
thisLowerLoss = lowerStandingLossCalc(params.Rval,lowerSA,lastLowerTemp,ambTemp);      % lower layer standing loss

thermConduc = ThermalConducCalc(ambTemp);        % thermal conductivity of thermocline at this time step - is dependant on water temp
layerThickness = 1e-3;                              % estimated value - (assumed to be infinitely small?) [m]

[topLoss, botGain] = thermoclineLoss(thermConduc,layerThickness,params.rad,lastUpperTemp,lastLowerTemp);    % heat exchanged across thermocline

usageLoss = upperUsageCalc(params.Cv,params.rho,thisFlow,lastUpperTemp,deltaT);       % heat leaving top of cylinder due to usage
% This should account for the energy in the cold water inserted into the lower layer.
coldFeedEnergy = coldInletCalc(params.Cv,params.rho,thisFlow,params.Tin,deltaT);

elementOutput = elementState * params.Q;
```

```
%%%%%%%%%%%%%%%%%%%%%% State Machine (working loop) %%%%%%%%%%%%%%%%%%%%%%

 switch(S(sC).nodeState)

       case 1             % fully mixed state

               upperChange = EdotUpperCalc(usageLoss,thisUpperLoss,topLoss);  % No thermocline or flow/draw loss
               % equate together sisCe this is the fully mixed state
               lowerChange = EdotLowerCalc(elementOutput,coldFeedEnergy,thisLowerLoss,botGain);


                newupperEnthalpy = totEnthalpyCalc(lastupperEnthalpy,lastlowerEnthalpy,upperChange,lowerChange,deltaT);
                 % Volume used here is total tank volume - this is temperature isCreas at first time step
                newTempMixed = tempCalc(newupperEnthalpy,params.Cv,params.rho,thisUpperVol);

                        if newTempMixed < params.Tin            % water temp can't go lower than inlet temp
                            newTempMixed = params.Tin;
                            newupperEnthalpy = baseEnthalpyCalc(params.Tin,params.Cv,params.rho,thisUpperVol);
                        end


% These are for the new node
S(sC).upperEnthalpy = newupperEnthalpy;
S(sC).Enthalpy = newupperEnthalpy;

S(sC).upperTemp = round(newTempMixed,1);
S(sC).lowerTemp = round(newTempMixed,1);

deltaJ = elementState*params.Q*deltaT*1;
S(sC).ArrCost = lastArrCost + deltaJ;
S(sC).elementState = elementState;

        %if elementState == 0     % Ensure costs don't match exactly for cases u = 0 vs u = 1

            S(sC).goCost = costGoHorz2(S(sC).timeStep,lastStep,finalTemp,params) + costGoVert(S(sC).Enthalpy,finalTemp,params) ;
            S(sC).totCost = S(sC).ArrCost + S(sC).goCost;    % should refer to pC node for arrive cost?


              if CLOSED_Hash.containsKey(S(sC).totCost) || CLOSED_Hash.containsValue(S(sC).structNumber)
              %continue
              S(sC).totCost = S(sC).totCost + costAddVal;     % tie-breaking in favour of elementState == 0
              end


case 2            % stratified state

     upperChange = EdotUpperCalc(usageLoss,thisUpperLoss,topLoss);
     lowerChange = EdotLowerCalc(elementOutput,coldFeedEnergy,thisLowerLoss,botGain);

     % Calculate energy for tank for first time step
     newupperEnthalpy = layerEnthalpyCalc(lastupperEnthalpy,upperChange,deltaT);
     newlowerEnthalpy = layerEnthalpyCalc(lastlowerEnthalpy,lowerChange,deltaT);

     newUpperTemp = round(tempCalc(newupperEnthalpy,params.Cv,params.rho,thisUpperVol),2);
     newLowerTemp = round(tempCalc(newlowerEnthalpy,params.Cv,params.rho,thisLowerVol),2);

      if newLowerTemp < params.Tin
          newLowerTemp = params.Tin;         % Water temp can't be lower than Tinlet
          newlowerEnthalpy = baseEnthalpyCalc(params.Tin,params.Cv,params.rho,thisLowerVol);
      end

     S(sC).upperEnthalpy = newupperEnthalpy;
     S(sC).lowerEnthalpy = newlowerEnthalpy;

     S(sC).upperTemp = round(newUpperTemp,1);
     S(sC).lowerTemp = round(newLowerTemp,1);

               deltaJ = elementState*params.Q*deltaT*1;      % Add electrity price here later?
               S(sC).ArrCost = lastArrCost + deltaJ;

           % Check for turnover event (bottom layer hotter than top layer)
           if newLowerTemp > newUpperTemp
               S(sC).nodeState = 1;
```

```
  else
      S(sC).nodeState = 2;

  end

   % Update the main record keeping arrays
   S(sC).elementState = elementState;


    S(sC).Enthalpy = newupperEnthalpy + newlowerEnthalpy;

% Ensure totCost isn't identical to any other nodes - can cause
% nodes to be repeated
  if elementState == 0
   S(sC).goCost = costGoHorz2(S(sC).timeStep,lastStep,finalTemp,params) + costGoVert(S(sC).Enthalpy,finalTemp,params);
   S(sC).totCost = S(sC).ArrCost + S(sC).goCost;      % should refer to pC node for arrive cost?


          if CLOSED_Hash.containsKey(S(sC).totCost) || CLOSED_Hash.containsValue(S(sC).structNumber)
              continue;
              %S(sC).totCost = S(sC).totCost + costAddVal;     % tie-breaking in favour of elementState == 0
          end
```

```
function [targetNodeCost, newStartNode] = maxTempNodeSearch(CLOSED_Hash,CLOSED_Time,CLOSED_Temp,restartStep)
%MAXTEMPNODESEARCH Determines whether any other nodes already exist at a given
%timestep, sorts these into an array and searches for the node with the hightest temperature.

%   This function's aim is to return the node struct. number for the node best placed to restart a new
%   heating profile curve.

nodeKeys = CLOSED_Hash.keySet.toArray;
nodeTimes = CLOSED_Time.values.toArray;
nodeTemps = CLOSED_Temp.values.toArray;

sameStepNodes_Key = zeros(1,length(nodeTemps));    % array to store nodes at the time step sC
sameStepNodes_Times = zeros(1,length(nodeTemps));
sameStepNodes_Temps = zeros(1,length(nodeTemps));

sizeNodeKeys = size(nodeTemps);

    for k = 1:sizeNodeKeys(1)

        if nodeTimes(k) == restartStep

            sameStepNodes_Key(k) = nodeKeys(k);
            sameStepNodes_Times(k) = nodeTimes(k);
            sameStepNodes_Temps(k) = nodeTemps(k);

        end


   end

   [maxTemp,maxIdx] = max(sameStepNodes_Temps);

   targetNodeCost = nodeKeys(maxIdx);

   newStartNode = CLOSED_Hash.get(targetNodeCost);
```

```
function heatingSteps = newHeatingCalc(timeStepSize,params,comfortTempMin,thisDraw)
%NEWHEATINGCALC Estimates number of timesteps required to heat a schedulted
%outlfow of water from Tin to comfortTempMin.
%  Based on rearragement of formula: E[kWh] = V[L].T[C]/860 Output is
%  min number of timesteps that water must be heated replace the energy
%  expected to be lost and prevent tank water temp from dipping below the
%  comfort threshold line.

% determine how much water needs to be replaced
% current time step found from current node (using S(sC))
steps = (thisDraw * (comfortTempMin - params.Tin)* 60 * 1000)/(0.001*params.Q*860);

heatingSteps = steps * 60/timeStepSize;   % actual number of timesteps to jump
end
```

2Node EHW model - starting parameters

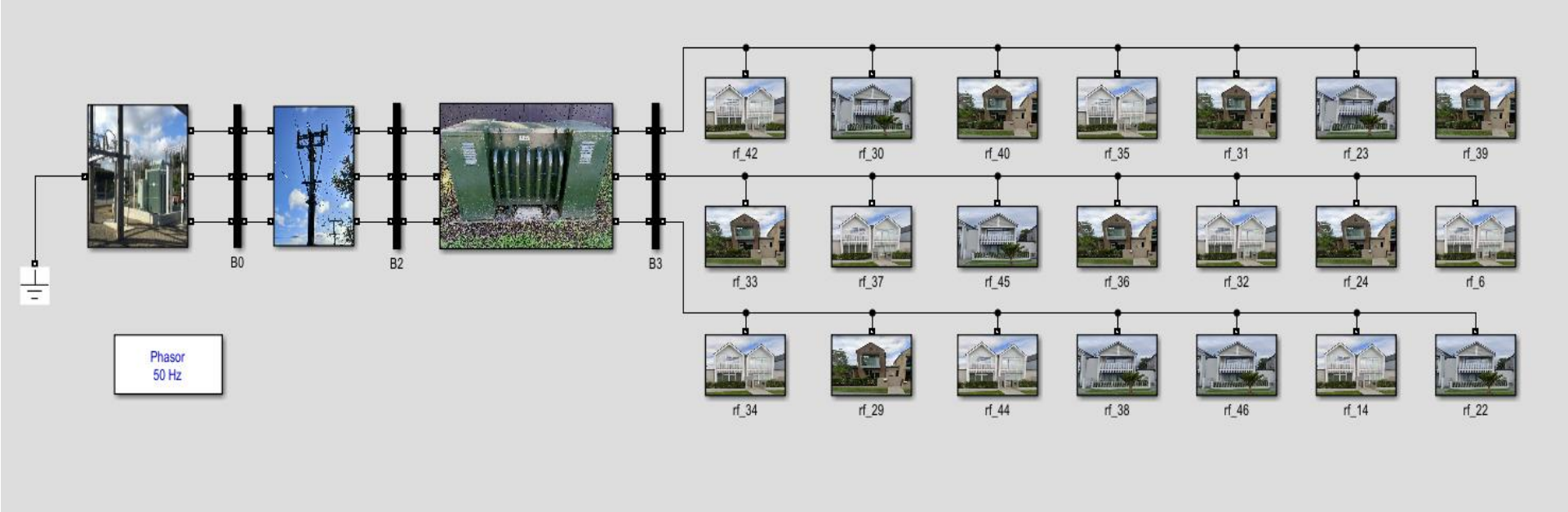
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Rvalue = 1;         % Typical NZ values 1-3 (M. W. Jack et al.)
U = 1/Rvalue;
Cv = 4180; %        % J/kg.K @ 15degC
rho = 999.2;        % kg/m^3
Q = 3000;

%Cylinder dimensions -
r = 0.2856 %0.56/2;  % m - from datasheet Peter Cocks
H = 0.39;    %100L
%H = 0.548;    %135L
%H = 0.6091;      %150L
%H = 0.72;   %180L
%H = 0.741;  %190L
%H = 1.2;    %300L

A = pi*r^2;
V = pi*r^2*H              % Volume [m^3]

% Starting Values
Tin = 12;   % Temperature of cold tap water (degC)
```

Create structure to export to model

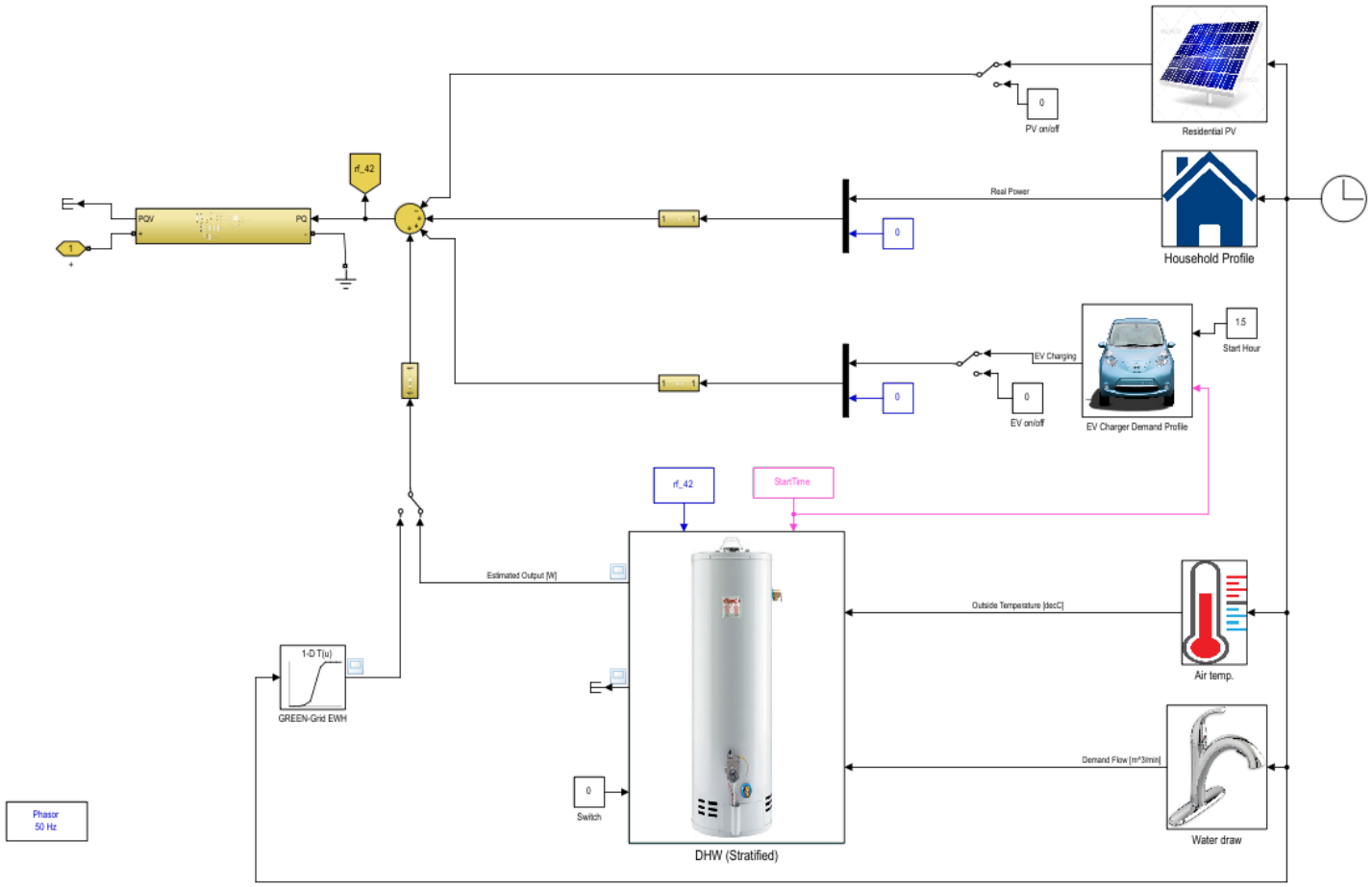
```
params.Rval = Rvalue;
params.U = U;
params.rad = r;          % radius of cylinder
params.area = A;           % Area of cylinder cross section
params.height = H;       % DHW cylinder height [m]
params.volume = V;            % DHW Tank volume [m^3]
params.mass = V*rho;        % Tank mass
params.Tin = Tin;        % Cold water inlet temp
params.rho = rho;        % mass density of water
params.Cv = Cv; % specific heat of water


params.Q = Q;            % heating element rating (W)

clear r Rvalue U UA rho Cv H Tin V
```

Appendix K: Simulink community microgrid model

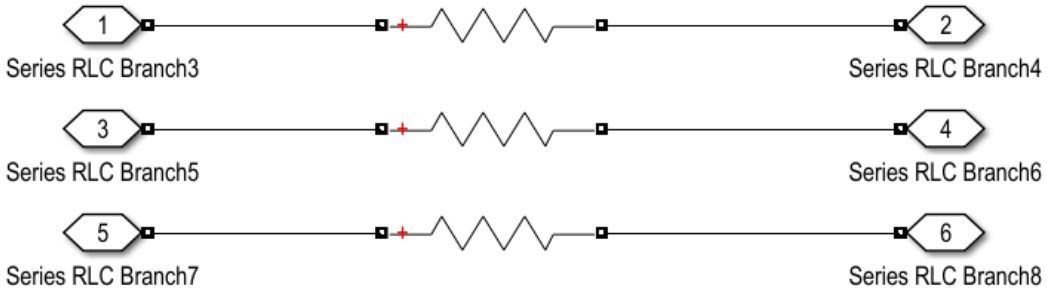
The Simulink model of a 21 House microgrid:



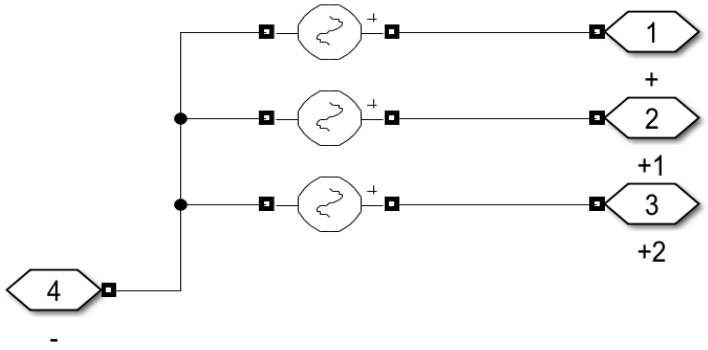
An exploded view of a representative Simulink model of households in the microgrid showing each component of the model, including the electric water heating cylinder.



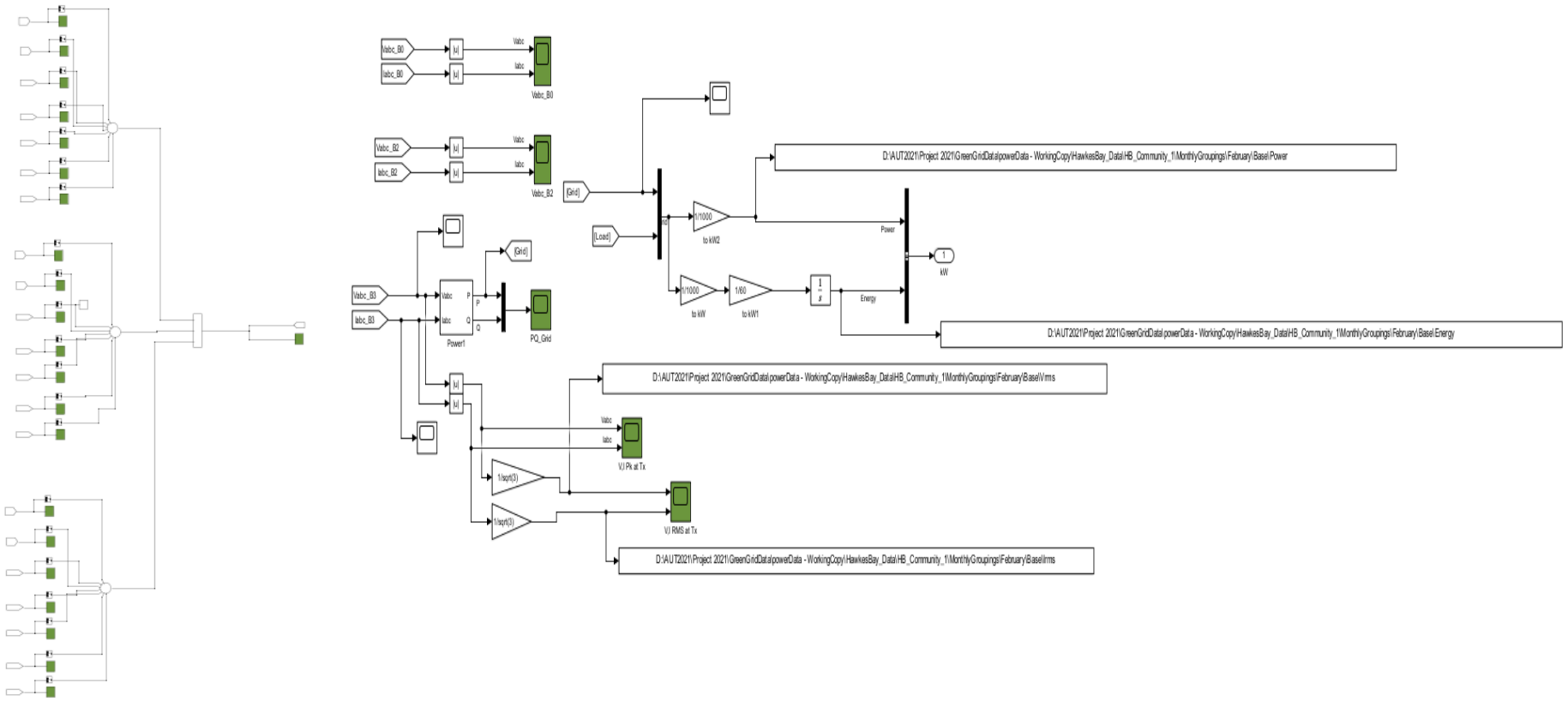
The Simulink model of the transmission lines between transformer and sub-station:



The Simulink model of the grid generation source:



Simulink code for collected scopes to visualise the results of the microgrid simulation:



Appendix L: EV Charge Scheduling Code

MILP control of multiple EV

```
% Optimization problem declared
EVopt = optimproblem('Description',"Optimise Multiple EV",'ObjectiveSense','minimize');


% time period comprised of 6 isON 30min segments = 12 total
numEV = 20;
numSteps = 48; % Adjusted to match otherDemand length
dt = 0.5; % Time step in hours (since we have 48 steps in 24 hours)
idxHr2ToEnd = 2:numSteps;


plot(JuneCommDemand_60min.Time(1:24),JuneCommDemand_60min.AggDemand(1:24))
xlabel('Time')
ylabel('Average Power (kW)')


maxGridPower = 2000;
```

```
% price of electricity at each time step
priceHourly = [0.2 0.2 0.2 0.2 0.2 0.2 0.3 0.3 0.3 0.2 0.2 0.2 0.2 0.2 0.2 0.2 0.3 0.3 0.3 0.3 0.3 0.2 0.2 0.2];
price = repelem(priceHourly, 2); % Repeat each price to cover 30-minute intervals


% forecast load profile (to come from simulation)
otherDemand = JuneCommDemand_60min.AggDemand(1:numSteps); % Ensure it has numSteps elements
targetProfile = maxGridPower * ones(numSteps,1);
profileToFill = targetProfile - otherDemand;


% there are 5 separate EV's that can be in binary state On or OFF at each time step

% draw of each EV heating element
Pnom = [7.4 7.4 7.4 7.4 7.4 3.7 7.4 3.7 7.4 3.7 7.4 3.7 7.4 3.7 7.4 3.7 7.4 3.7 7.4 3.7];
PnomArray = repmat(Pnom, numSteps, 1);
Pmin = zeros(1, numEV);
PminArray = repmat(Pmin, numSteps, 1);
```

```
isON = optimvar("isON", numSteps, numEV, "Type", "integer", "LowerBound", 0, "UpperBound", 1);
powerDraw = optimvar("powerDraw", numSteps, numEV, 'LowerBound', 0, 'UpperBound', PnomArray);

startup = optimvar('startup',numSteps,numEV,'Type','integer','LowerBound',0,'UpperBound',1);

powerCostArray = sum(powerDraw'*price',1);
isONcost = sum(isON'*price',1);
        %sum(powerDraw*powerCostArray,1);
EVopt.Objective = sum(sum(powerDraw .* (price' * dt)));

% show(costOpt);
```

```
minTimeOn = 1*ones(1,numEV);
maxOFFTime = 48*ones(1,numEV); %is acting as min time ON

EVopt.Constraints.powerUpperBound = powerDraw <= PnomArray .* isON;
EVopt.Constraints.powerZeroWhenOff = powerDraw >= 0; % Redundant but clarifies intent

EVopt.Constraints.gridLimit = sum(powerDraw, 2) <= profileToFill;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Solution - intlinprog
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

options = optimoptions('intlinprog', 'Display', 'iter');

[sol, optvals] = solve(EVopt, "Options", options);

% Display solution variables
disp(sol.isON);
disp(sol.powerDraw);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% Plot solutions - visualisation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Adjust indices and data sizes in plotting
hf = figure('visible','off','units','pixel','position',[100 100 760 800]);
movegui(hf,'center');
set(hf,'visible','on');

% Schedule Matrix
subplot(3,1,1);
imagesc(0.5:numSteps, 1:numEV, sol.isON');
set(gca, 'xtick', 1:2:numSteps, 'ytick', 1:numEV, 'yticklabel', labels);
ylabel('EV Number', 'FontSize', 12);
title('EV Charging Schedule');

% Power Draw Matrix
subplot(3,1,2);
imagesc(0.5:numSteps, 1:numEV, sol.powerDraw');
set(gca, 'xtick', 1:2:numSteps, 'ytick', 1:numEV, 'yticklabel', labels);
ylabel('EV Number', 'FontSize', 12);
title('EV Power Draw');

% Total Power Demand
hourlyTotalEV = sum(sol.powerDraw, 2);
totalDemand = hourlyTotalEV + otherDemand;
subplot(3,1,3);
plot(1:numSteps, otherDemand, 'LineWidth', 2);
hold on;
plot(1:numSteps, totalDemand, '--k', 'LineWidth', 1);
bar(hourlyTotalEV, 'FaceAlpha', 0.5);
xlabel('Time Step');
ylabel('Power (kW)');
legend('Other Demand', 'Total Demand', 'EV Demand');
title('Power Demand Over Time');
```