UNIVERSITY OF NAIROBI

Faculty of Engineering

Design and Testing of a Demand Response Q-Learning Algorithm for a Smart Home Energy Management System

By

Walter Angano

(F56/33270/2019)

A research project submitted in partial fulfillment for the Degree of Master of Science in Energy Management in the Department of Mechanical and Manufacturing Engineering in the University of Nairobi

April 2021

Approval

This research project has been submitted for examination with our knowledge and approval as University supervisors.

Prof. Wekesa Cyrus	Signature	……………………..	Date	12/07/2021…………………
Dr. Musau Peter Moses	Signature	……………………..	Date	12/07/2021…………………

Dedication

I dedicate this research to Gladys Musimbi, Evans Adonda, Margaret Andeso, Atkins Kamave, Kevin Angano, Erick Angoya and Paul Oyiengo. Indeed their support and encouragement promoted an active research work environment. Besides, much dedication to the Late Musa Angano and Bethlina Jakoyo, as my passion to pursue higher education stems from their legacy.

Acknowledgment

I would like to thank my research supervisors, Professor Wekesa Cyrus and Dr. Musau Peter Moses, for their invaluable and absolute professional guidance in the course of this research. Special thanks to Virunga Power Team for their support towards realizing this research.

Declaration of Originality

This Report is my original work and has not been presented for a degree award in any other Institution for degree award or other qualification.

Name of student: Walter Angano

Registration: F56/33270/2019

Faculty: Engineering

Department: Mechanical and Manufacturing Engineering

Course Name: Master of Science (Energy Management)

Title of work: Design and Testing of a Demand Response Q-Learning Algorithm for a Smart Home Energy Management System

1) I understand what plagiarism is and I'm aware of the university policy in this regard
2) I declare that this research proposal is my original work and has not been submitted elsewhere for examination, award of a degree or publication. Where other works or my own work has been used, this has properly been acknowledged and referenced in accordance with the University of Nairobi's requirements
3) I have not sought or used the services of any professional agencies to produce this work
4) I have not allowed, and shall not allow anyone to copy my work with the intention of passing it off as his/her work
5) I understand that any false claim in respect of this work shall result in disciplinary action in accordance with University of Nairobi anti-plagiarism policy

Name: Walter Angano Adm. No: F56/33270/2019

Signature: .... **Date:** 1st September 2021

Design and Testing of a Demand Response Q-Learning Algorithm for a Smart Home Energy Management System

Walter Angano (F56/33270/2019)

Abstract

Growth in energy demand stimulates a need to meet this demand which may be achieved either through wired solutions involving infrastructural investment in generation, transmission and distribution systems or non-wired solutions such as demand response (DR). DR is a grid load reduction measure in response to supply constraints where consumers voluntarily participate in shifting their energy usage during peak periods in response to a time or price-based incentive. In Kenya, residential consumers constitute approximately 33 while 30-40 percent on a global scale which demonstrates an essential fraction for their participation in DR. This research aimed at reviewing smart home energy management systems, Reinforcement Learning (RL) techniques such as Q-learning, designing and testing a single agent Q-Learning algorithm to objectively determine an optimal policy from a set of load management strategies. The study sought to address the performance of the algorithm by reducing the learning speed of the agent. This was achieved by introducing a continuous knowledge base that updated fuzzy logic rules and setting up a definite state-action space. The algorithm was implemented in Matlab and interfaced with the physical environment using the Arduino Uno kit while adopting serial communication between simulation and physical environment. A graphical user interface developed using the app designer tool in Matlab created a provision for integrating consumer feedback which was critical in communicating with the knowledge base to update fuzzy rules. The Time of Use (ToU) tariff plan constituted three major segments which were off-peak, mid-peak and peak tariffs, developed by benchmarking public historical residential tariff data with ToU trends for other countries. Load profiles generated from appliance and ToU data were used to test the algorithm. The designed algorithm showed an improvement in learning within 500 episodes and net energy savings ranging between 8 and 11 %.

Keywords: Demand Response (DR), Reinforcement Learning (RL), Smart Home Energy Management System (SHEMS), Matlab, Arduino Uno, Time of Use (ToU)

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Definition
COD	Cut-off Date
CPP	Critical Peak Pricing
DG	Distributed Generation
DHW	Domestic Hot Water
DR	Demand Response
DSM	Demand Side Management
EV	Electric Vehicle
GUI	Graphical User Interface
HEMS	Home Energy Management System
HVAC	Heating, Ventilation and Air-Conditioning
KPLC	Kenya Power & Lighting Company
LC	Load Clipping
LCL	Load Control Level
LCPDP	Least Cost Power Development Plan
Matlab	MATLAB Version: 9.0.0.341360 (R2016a)
MDP	Markov Decision Processes
MW	MegaWatt
NO	Normally Open
PPA	Power Purchase Agreement
RL	Reinforcement Learning
RTP	Real-time pricing
SHEMS	Smart Home Energy Management System

Chapter 1 Introduction

1.1 Background

Demand Side Management (DSM) includes programs by utilities that encourage energy consumers to be energy efficient and considered long-term. As a vehicle of DSM, Demand Response (DR) refers to short-term responses to electricity market prices expressed by utilities [1]. It includes programs developed for end-users to perform short-term load reductions when the energy market pricing is high particularly during peak hours. Such programs are applied to smart end-user systems whose infrastructure is integrated with smart grids through internet-based communication. Internet-based communication enables the smart grid to implement certain DSM measures targeted to modify smart end-users energy demand while smart energy consumers voluntarily respond to the utility's package request.

There is a distinction between DSM and DR. DSM is implemented purposely to reduce peak electricity demand and defer high capital investment in constructing the generation, transmission, and distribution infrastructure that would have been necessary to offset the demand. This is done by managing energy demand and supply by utilities. DR aims at reducing the energy costs for a particular incentive. DR Programs are meant to incentivize energy consumers and are categorized into price-based and incentive-based [2]. Price-based are more appropriate for residential consumers while incentive-based are suited for industrial consumers. Price-based DR schemes divide a day into several time blocks and corresponding electricity prices that reflect actual electricity market prices.

The United States implemented its first large-scale residential Real-Time Pricing (RTP) program, referred to as the Energy-Smart Pricing Plan SM (ESPP) program. According to [3], the ESPP program integrated a feature of a day-ahead notification where consumers are notified through email or telephone or websites. Another aspect is the program provided information about energy usage, instructions on how to reduce usage during peak periods. Another feature is the Price Light consisting of a color-changing small globe. The globe is assigned a color-coding to represent the electricity tariff for that hour. One of the key findings of the ESPP program is RTP can effectively establish a demand-side management program. Also, the need for automation could make the entire process a success at both high and low prices while considering consumer's preferences.

There has been an evolution in the scientific publications on algorithms that implement DR in Smart Home Energy Management System with Reinforcement Learning (RL) being the predominant method. However, most RL algorithms have been tested in simulation environments with limited testing in physical systems while others presented approaches that are considered complex for a simple residential system.

1.2 Problem Statement

Electrical energy has the advantage of versatility (can be put to multiple uses), cleanliness and can be transported at the speed of light. However, one major problem this form of energy faces is the expense of providing grid-scale storage. For this reason, the energy generated must simultaneously be consumed. That is, energy generation must balance energy demand plus energy losses at all times, a necessity that also facilitates support for system integrity (constancy of system frequency).

One of the tools for balancing demand and supply is the activation of demand response (DR) mechanisms, involving the engagement of customers to modify their energy consumption so that peak demand is reduced. This is seen as an often more effective option than expanding generation infrastructure to meet the peak demand or even occasional demand spikes. In this case, DR becomes a critical resource for the operations of the power grid. It is also a resource through which the customer can reduce energy bills. Effective DR, from the customer perspective, depends on attractive electricity price signals. However, in Kenya and many other countries, customers pay the same electricity price irrespective of the time of use. Also, previous algorithms have been faced with a curse of dimensionality (learning agent has lower learning speeds) and are limited to simulation environments. Furthermore, no study has been undertaken to establish how customers will respond to varying electricity pricing or even to determine if demand response mechanisms would be invaluable.

This study sought to address these gaps by focusing on the design and testing of a DR algorithm in a smart home energy management system (SHEMS) targeting household consumers.

1.3 Justification

In the Kenyan context, Kenya Power & Lighting Company (KPLC) kicked off the implementation of the time of use rate (ToU) primarily focusing on the commercial and industrial segment. The ToU scheme provides a fifty percent discount on the energy charge rate upon attaining a consumption threshold. The implementation of the ToU realized energy sales totaling 91 GWh corresponding to US$ 4 million in additional revenue [4]. Kenya's Least Cost Power Development Plan (LCPDP) outlined recommendations towards ensuring effective implementation of generation and transmission expansion plans [5]. This included the introduction of ToU tariffs in domestic systems and the reintroduction of interruptible tariffs for use that promote balancing household consumption.

The introduction of the lifeline tariff was to stimulate energy use for households located in informal settlements, peri-urban and rural areas. This tariff caps energy usage at 10 kWh at a charge rate of US$0.1089 per KWh and is meant to protect low-income households, who constitute approximately 5.7 million consumers, from the high living cost and affordable energy for basic use such as lighting, charging phones. The ordinary tariff charge rate is US$ 0.1425 per kWh for units above 10 and below 15,000 with over 2.5 million domestic consumers [6].

A significant amount of cost could be saved and energy demand and supply balance envisioned by introducing voluntary ToU tariffs for the DC ordinary category particularly for consumers with peak usage and who are flexible in shifting their demand to off-peak times. This could avoid an energy economic crisis due to excess generation similar to the situation experienced by Ghana's public utility [7]. A possible interpretation of the findings of Kenya's LCPDP is Kenya may have an excess generation in the future. DR encourages consumers of a Smart Home Energy Management System (SHEMS) to respond effectively to energy market prices purposely to enhance grid balance and energy bill reduction. This offsets the need for generation, transmission and distribution infrastructural expansion.

There is limited research that focuses on demand response in the Kenyan market. Besides, the author [8] has established significant research gaps in previous related works. One of the gaps includes limited testing of proposed algorithms on physical systems. While recommendations of Kenya's long-term least development plan include the introduction of ToU at the residential level, it is essential to conduct research studies on DR programs that emphasize voluntary participation into optimal energy usage by integrating the input of domestic consumers and improving robustness in SHEMS.

1.4 Research Objectives

1.4.1 Main Objective

The main objective of the research was to design and test a demand response Q-learning algorithm for a Smart Home Energy Management System (SHEMS).

1.4.2 Specific Objectives

The specific objectives are:

i. Review of algorithms applied in demand response (DR), smart home energy management systems (SHEMS) and Time of Use (ToU) tariff structures.

ii. Generate fuzzy logic control rules to estimate rewards for load management strategies.

iii. Develop a Q-learning algorithm, based on fuzzy rules and ToU, to establish optimal action from a set of actions.

iv. Test the Q-learning algorithm in an Arduino Uno.

1.5 Research Question

The research anticipated responding to the following research questions.

i. What is the research trend of demand response algorithms including their integration in smart home energy management systems and Time of Use tariff?

ii. What is the impact of fuzzy logic control on the learning speed of the algorithm?

iii. What is the learning speed of a Q-learning algorithm when integrated with continuous improvement fuzzy control rules?

iv. What is the robustness of Arduino Uno in testing the algorithm?

1.6 Scope

The research scope entails a literature review of previous demand response algorithms to establish research gaps, formulation of demand response problem by setting up an objective function, developing a research framework, designing and testing a Q-learning algorithm in Matlab and Arduino Uno.

1.7 Report Organization

This research report is organized as follows: Chapter 1 describes the background and justification of the research, the purpose for the proposed research, and its scope and objectives; Chapter 2 reviews the existing literature on predominant algorithms, establishes research gaps, their extent and research limitations; Chapter 3 provides the framework for formulating the demand response problem in a residential smart system and addressing research gaps; Chapter 4 Provides the findings of the research and its interpretation; Chapter 5 presents the detailed recommendations deduced from the findings of the research.

Chapter 2 Literature Review

2.1 Introduction to Energy Management

Energy management is "the proactive, organized and systematic coordination of procurement, conversion, distribution and use of energy to meet the requirements, taking into account environmental and economic objectives" [9]. Energy management includes features ranging from behavioral changes, better operation & maintenance, energy-efficient retrofits, energy recovery, fuel consumption to temporary and permanent peak demand reductions, and distributed generation systems. These features constitute demand-side management, load management, demand response, energy efficiency, fuel switching and distributed energy resources as described below [10], [11], [12].

Demand-side management (DSM) is the planning, execution and monitoring of utility activities to achieve the desired utility's load shape. It includes managing all forms of energy on the demand side and features the following (a) influences consumer's energy use (b) objective-oriented spanning from improvement in customer satisfaction to achieving reliability targets and (c) identifies consumer's response to programs (d) its value is influenced by load shape.

Load management is a subset of DSM and is defined as actions taken by utilities to interfere with the load that is visible to their generating systems to achieve optimal and economic operating conditions. The utilities are concerned with improving their load factor and reducing peak demand. Examples of load management strategies include

- Peak clipping (load shedding) refers to measures that reduce a system's peak load demand by utilities' direct control of consumer loads.
- Valley filling refers to increasing loads during off-peak times to encourage energy consumption of the surplus capacity for example electric vehicle charging during off-peak times.
- Load shifting refers to shifting loads from peak to off-peak periods. An example is storage strategies for space and water heating.

Demand response is considered as a subset of load management and refers to voluntary actions by the consumer to reduce the load in response to time or incentive-based signals.

Energy efficiency (EE) is the adoption of alternative processes, equipment, techniques that produce the desired output with less energy. EE targets to meet energy needs with less energy by improving the productivity of energy resources.

Fuel switching entails a substitution of an energy source for another driven by fuel cost-saving, environmental regulations, saving scarce resources, or agreement with a fuel supplier. Distributed energy resources (DER) are technologies for decentralized generation, storage and power quality. DER can be applied at a local (building) or utility-level feeding into the distribution grid reducing transmission and distribution losses.

There is bipartite global awareness of energy management and climate both in developed and developing nations. The first awareness is the critical role energy plays in national economies and the second is understanding that global warming is due to increasing cumulative emissions of heat-trapping particles and gases emitted by internal combustion engines and thermal power plants. The industrial revolution led to the discovery and exploitation of fossil fuels and since then the global economy has largely been powered by fossil fuels. According to [13], the global warming trend is majorly due to an increase in Carbon (IV) oxide (CO_2) since 1880 as illustrated in Figure 2-1.

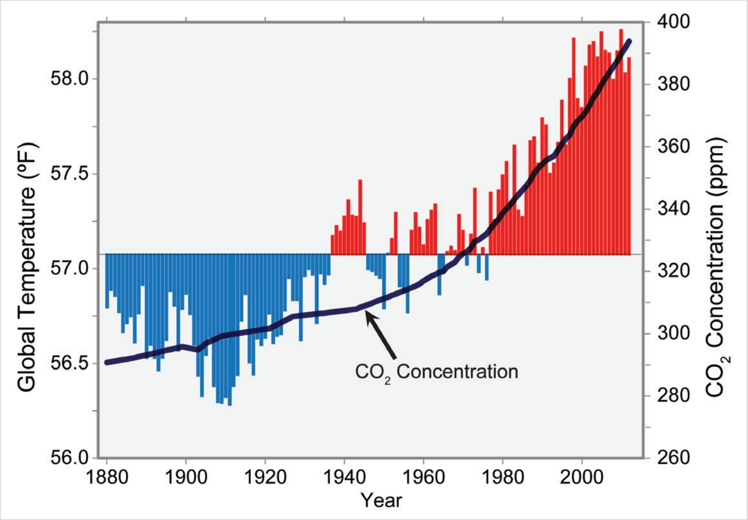


Figure 2-1Global temperature and carbon dioxide trend since 1880 [13].

Figure 2-1 shows that unlike temperatures (red and blue histogram) which are significantly normalized or fluctuate to natural processes such as volcanic eruptions, CO_2 concentration increases independent of natural processes.

Stringent global environmental quality standards are now at an advanced phase with industries, governments and other players facing the pressure for environmental quality compliance. Economic competitiveness and complying with increasing environmental standards are some of the key drivers in making decisions for capital cost investments by organizations. Energy management is a tool that enables organizations to meet such objectives for both short and long-term success.

The authors in [14] outline energy management benefits to include dramatic reduction in the amount of CO_2 emissions by a reduction in the combustion of fossil fuels hence reduce global warming. This is achieved by reducing the load that needs to be served by fossil-fuel power plants or deferring them to renewable energy sources, therefore less thermal pollution by fossil-fuel power plants.

2.2 Demand Response

2.2.1 Demand Response Schemes

The two broad demand response schemes are incentive (or event-based) and price-based. According to [15], [16], [17] incentive (event-based schemes or program options) are voluntary designed programs in which consumers are rewarded for reducing their loads upon request or offering a utility some level of control over some appliances to reduce energy consumption during peak demand while price-based scheme or tariff option is dynamically designed to flatten the demand curve by motivating customers to change consumption patterns through time-varying electricity prices. Examples of price-based programs include the Static Time of Use (ToU) rates, Critical peak pricing (CPP), and Real-Time Pricing (RTP).

Incentive-based demand response schemes include

- Direct load control schemes are programs where the utility shuts down or cycles a consumer's electrical equipment on short notice.
- Demand bidding/buyback schemes are penalty-based programs that encourage heavy customers to bid into a wholesale electricity market and offer to provide load reductions at a price at which they are willing to be curtailed, or identify how much load they would be willing to curtail at a utility-posted price.
- An interruptible/curtailable scheme includes penalty-based programs, integrated with the customer tariff that provides a rate discount or bill credit for agreeing to reduce load, typically to a pre-specified firm service level (FSL), during system contingencies.

- Capacity market schemes are both day-ahead and penalty-based programs typically offered to customers that can commit to providing pre-specified load reductions when system contingencies arise.
- Emergency demand schemes are programs that provide incentive payments to customers for measured load reductions during reliability-triggered events; emergency demand response programs may or may not levy penalties when enrolled customers do not respond.

Price-based demand response schemes (sometimes referred to as Time of Use Tariffs) include

- Static Time-of-use (ToU) pricing scheme determines tariffs in advance and prices are usually defined for a 24-hr day over large usage blocks. Tariffs are determined in advance. Countries, where the scheme is common, include Europe and Italy. Rates may simply be night and day pricing at peak or off-peak times.
- Dynamic real-time pricing scheme where the price of electricity fluctuates hourly reflecting the real-time conditions in the wholesale price of electricity.
- Critical peak pricing schemes involving pre-specifying high rates for usage at critical peak periods. It is a combination of static and dynamic pricing.

2.2.2 Benefits of Demand Response

The overarching benefit of demand response described in [16] is an improved resource efficiency in energy generation, transmission and distribution. Improved resource efficiency provides additional benefits which can be categorized into four main categories. These are

- Participant financial benefits realized in form of incentive payments and bill savings which are consequent of consumer's participation in demand adjustment in response to incentive and price-based programs respectively.
- Market financial benefits are realized by avoidance of the need for peaking power plants. Production costs and therefore energy prices at wholesale purchase are significantly reduced. Overall, it results in tariff reduction, capacity increase and deferred new power plant infrastructural costs.
- Reliability benefits are realized as reduced forced outages that otherwise lead to imposed financial costs and consumer dissatisfaction.
- Market performance benefits realized as mitigation to power supplier's monopoly on raising energy prices beyond production costs.

2.3 Fundamental Components and architecture of a HEMS

In-home HEMS infrastructure consists of the communication network, smart meters, HEMS center, and smart appliances [18] [19] [20] [21]. Figure 2-2 illustrates the components and functionalities of a SHEMS.

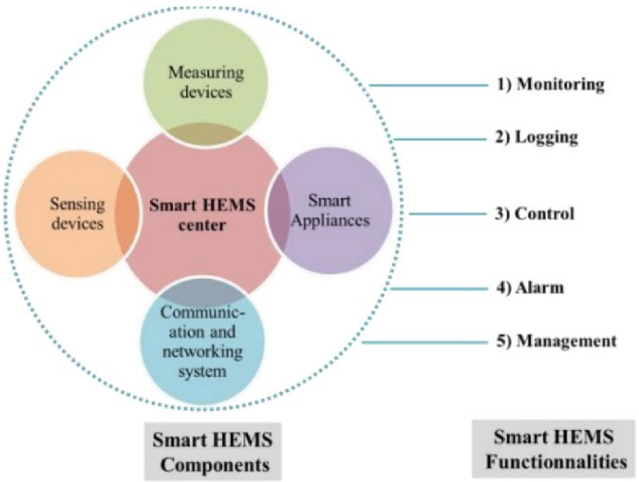


Figure 2-2 Components and functionalities of a SHEMS [21]

Communication system/enabling ICT interconnects several appliances, sensors, displays, Renewable Energy Storage (RES), and Electric Vehicles (EVs). Wireless and wired communication technologies such as Wi-Fi, Home plug, Z-wave, and ZigBee exist.

Smart appliances include domestic appliances and energy storage devices. They are classified into schedulable and non-schedulable appliances. Non-schedulable appliances include lights, printing machines, television, dryers and microwaves. Schedulable appliances may include water heaters, iron, washing machines, EVs. Depending on their continuity of time, schedulable appliances can be further classified as interruptible schedulable appliances such as vacuum cleaners, water heaters, humidifiers and non-interruptible schedulable appliances such as washing machines, clothes dryers. Schedulable appliances can be switched on and off, or scheduled for optimal operation.

Smart HEMS center is the core of the HEMS and provides monitoring, logging, control, management, and alarm module functionalities. Sensing devices for HEMS include current, voltage, temperature, motion, and light sensors. Measuring devices for data measurement and transmittal.

2.4 Introduction to Machine Learning

Machine learning entails developing computer models that learn, modify, or adapt their actions accurately without following explicit instructions. Algorithms are used to train computer models by extracting relationships or useful information from a massive dataset [22]. Machine learning algorithms can be classified into supervised, unsupervised and reinforcement learning as illustrated in Figure 2-3, depending on the algorithm training method.

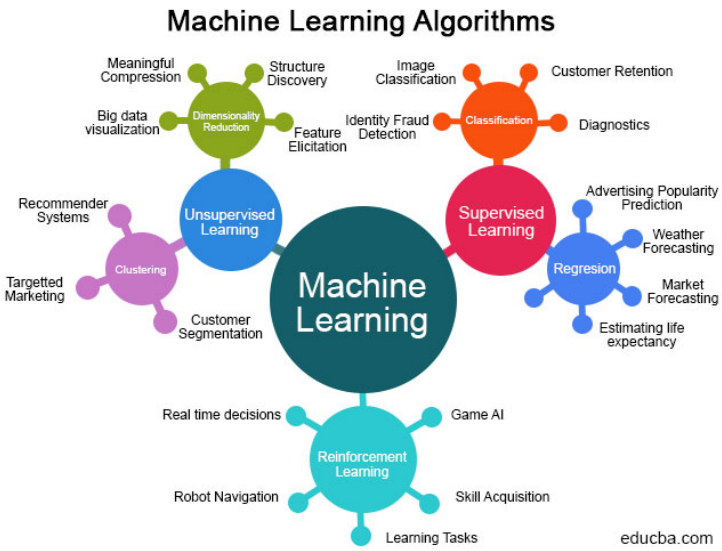


Figure 2-3 Machine learning algorithm classification [22].

- In Supervised Learning, the training dataset is labeled. This means each feature or independent variable and its desired corresponding target for training the model is provided. The algorithm trained based on the dataset responds to all possible inputs. Regression and classification techniques are used.
- In unsupervised learning, the training dataset is not labeled therefore the algorithm identifies resemblances in the input data and through techniques such as dimensionality reduction and clustering categorizes the data into similar groups.
- Reinforcement learning combines the trade-offs in supervised and unsupervised learning to eliminate the limitations featured in each. The algorithm explores a range of possibilities until it determines how to get the right response. The algorithm learns what to do to maximize a numerical rewards signal. The learner/agent discovers which actions yield optimal reward by trying those actions.

2.5 Reinforcement Learning and its Application

2.5.1 Introduction to Reinforcement Learning

Reinforcement learning (RL) is a branch of machine learning that entails sequential decision-making to achieve the desired goal. RL problem constitutes the components illustrated in Figure 2-4. RL problem when formulated using a mathematical framework such as a Markov Decision Processes (MDP) entails an agent-environment interaction, where actions influence subsequent steps and results.

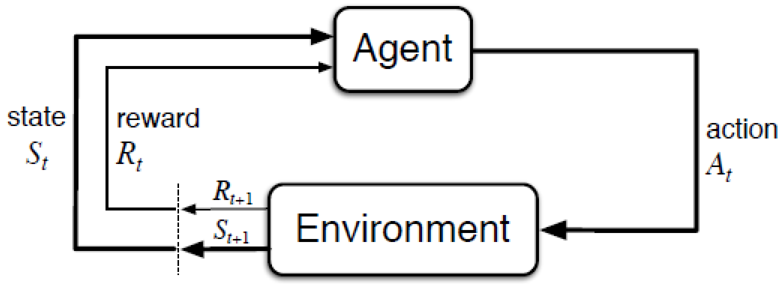


Figure 2-4 Reinforcement learning cycle [23] and [24].

An agent is a controller and decision-maker who continually interacts with the environment at each discrete time *step t* in form of action at each of a sequence of discrete-time steps. The agent receives the environment's state $S_t \epsilon\ S$. The next step is the agent selects an action $A_t \epsilon A(s)$ and sends it to the environment. The agent receives feedback on a numerical reward one-time step later, R_{t+1} based on a consequence of its action and a new state S_{t+1}.

An environment/ controlled system is a physical system with which the agent interacts. The environment responds to the agent with a new state and reward as a consequence of the previous action.

2.5.2 Elements of Reinforcement Learning

From the previous section, the ultimate objective of the agent is to maximize the sum of rewards for the corresponding actions taken. Given a set of actions applicable to a particular state, the agent is required to select an optimal action and this is done by a policy. The cumulative reward that can be achieved from a state is referred to as the value function. The concepts of RL include [23];

- A policy can be described as how an agent selects an optimal action from a set of actions in a given state that maximizes the cumulative rewards. A policy can be deterministic or stochastic. Deterministic policies provide a single optimal action that the agent needs to take while stochastic policies provide probability distribution for the actions in a state.
- Value function represents the cumulative reward anticipated in the future assuming an agent starts in a given state. Value function can be a state-value function (a function of state) or action-value (a function of state-action pair). Action-value methods (one of the three main families of RL algorithms) learn an action-value function to select an optimal action. The other two methods include policy-gradient and actor-critic methods.
- Rewards are numerical values that the agent seeks to maximize throughout the set of actions. Reward reflects how the action was to the agent, whether good or bad.

2.5.3 Applications of Reinforcement Learning

Reinforcement learning has been applied in diverse fields such as autonomous machines (smart grids, robotics), control (quality control, fault system detection), and optimization problems (supply chains and process planning). In optimization problems, RL is applied in energy optimization and smart grids where it can be used to adjust electricity demand in response to time-based or incentive-based programs or reduce energy usage [23].

A review of related research works demonstrates reinforcement learning (RL) as a tool for formulating demand response such as optimal scheduling and energy consumption by appliances, consumer energy trading with the grid to maximizing consumer comfort.

2.6 Related Research Works

This section describes the trend in publication and research gaps of DR in residential systems.

2.6.1 Previous Review of Algorithms and Modelling Techniques

A review of DR algorithms and modeling techniques by [8] illustrates Reinforcement Learning (RL) as a predominantly applied method in DR applications. RL is considered an agent-based Artificial Intelligence (AI) algorithm, characterized by the capability and adaptability to learn user preferences through interaction and is model-free. RL algorithm is considered more suitable in real-world applications, particularly DR.

The authors extensively reviewed the literature on algorithms and modeling techniques up to the date of 10/23/2018 (cutoff date). The author’s classification of the articles by variant of the adopted RL algorithm and energy systems being controlled is summarized in Table 2-1. Q-learning is the most widely adopted technique in energy systems control irrespective of the application area. The application areas include Heating Ventilation and Air-conditioning (HVAC) systems, Domestic Hot Water (DHW), smart appliances, Electric vehicles & HEV, Distributed Generation (DG) & Storage.

The authors flagged research gaps and proposed paths that are relevant for consideration in future research. The gaps provide the basis of investigation and conducting an additional literature trend between the cutoff date by the authors and this research. The relevant gaps flagged and which form the basis of additional literature review include

- Most reinforcement learning algorithms have been performed in a simulation environment which has limited the implementation of such algorithms in residential and commercial buildings. Testing of algorithms in physical systems is a potential research path to measure the capability, flexibility and reliability of control by reinforcement learning agents.
- Few publications considered human feedback through estimation of dissatisfaction function. Incorporation of actual human feedback in RL testing facility would measure the effectiveness of algorithms.
- Some algorithms are characterized with a curse of dimensionality problem particularly for large state-action where the speed of convergence is significantly reduced and subsequently learning speed by RL agent. The authors proposed data preprocessing to extract relevant features to reduce the curse of dimensionality. Data augmentation algorithms such as Generative Adversarial Neural Network could help in improving the data sets relevant for the training process.

The authors recommended a framework to outline the critical definition of parameters to enhance understanding and ease of reproducing similar results by readers. The parameters derived from the review and relevant to this research is to specify whether;

- Electricity prices are modeled as demand independent or dependent and if it is deterministic or stochastic;
- States, actions, and rewards are modeled as deterministic or stochastic;

- Demand response algorithm is model-based or model-free, on-policy, or off-policy;
- An actual or estimated human feedback was used and testing of the algorithm was in a simulation environment or a physical system.

The subsequent subsections review the relevant literature on demand response algorithms for smart residential energy management systems to determine whether outlined gaps above were addressed or establish potential new research gaps which are critical in developing the problem formulation.

Table 2-1 Summary of publication on demand response until the cut-off date [8]

	No of Publications per energy system			
RL Algorithm	HVAC & DHW	Appliances	EV & HEV	DG & Storage
Q-learning	17	7	19	14
Q(λ)	1	1	1	
Fuzzy Q-learning				1
Sarsa				
BRL				2
TD	1		1	
TD(0)			1	
W-learning			5	
Monte-Carlo	1		1	
Policy iteration	1			
Actor critic	4	1		1
Dual Q-learning				2
Multi-player RL		1		

2.6.2 Community-Based Energy Management based with Artificial Intelligence

A community-based energy management system consisting of an energy pool, intelligent and non-intelligent domestic consumers is proposed in [25]. The authors incorporated a trading concept where intelligent and non-intelligent domestic consumers could trade in an energy pool. Intelligent consumer systems with Energy Storage Systems (ESS) participated in arbitrage trading (Consumers sell surplus amounts of energy to the energy pool and buy back at real-time pricing) while non-intelligent consumer systems had the option to purchase energy from the energy pool at a price relatively lower than the market price.

Intelligent domestic homes have a smart agent that manages the ESS based on the pool pricing and neighborhood energy demand. The authors modeled the problem as an MDP and RL (Q-learning) used to select an optimal decision. The MDP framework constituted a set of states (state of charge, real-time retail electricity price, and community market price). A set of actions for electricity amount (the decision to buy or sell energy) and storage (charge or discharge energy storage systems) by intelligent users, dimension matrix Q-value table for recording the Q value for the corresponding actions, a decision-making agent that selects an action with the maximum Q-value on the Q-value table and reward function which awards the agent with the reward for an action (storage and electricity amount).

The ESS and price model are considered to be continuous and the impact is an infinite state and as such no optimal action would be found. The authors preferred a fuzzy inference system for the battery and price by setting a fuzzy logic where values of input vector through fuzzy rules are translated into corresponding output vector. The fuzzy rules represent the infinite states of energy price and SOC as finite states. The Q-learning model was validated using data from case studies. The author concluded RL is an effective method for solving MDP-type problems. Fuzzy systems can provide finite approximations of the infinite process making Q-learning possible in solving infinite systems.

2.6.3 Energy Management of a Smart Home Reinforcement Learning

A data-driven multi-agent RL approach is proposed to ensure optimal energy consumption of appliances to minimize energy bills and discomfort cost within the equality and inequality constraints of consumer Air Conditioning (AC) thermal comfort setting, operations of appliances, and distributed energy system [26]. The study focuses more on consumer thermal discomfort review. Q-learning algorithm function includes scheduling of appliances, charging and discharging of Energy Storage System (ESS). Appliances are categorized into two controllable (reducible or shiftable – interruptible and non-interruptible) and uncontrollable.

In formulating the objective as an MDP problem, the authors' objective function is the total cost of controllable and uncontrollable appliances and dissatisfaction cost difference of preferred and indoor temperature. The state space is a 24-hr period at 1h schedule resolution while the action space varies binary action for shiftable non-interruptible loads such as washing machine, discretized energy consumption with various levels for thermostatically controllable loads such as AC. The reward system is the summation of negative energy costs and negative dissatisfaction costs (thermal discomfort, undesired operation, under or overcharging of ESS) linked to consumer preference. Q-learning agent independently establishes an optimal policy to reduce energy costs within the range of consumer comfort level.

Artificial Neural Network (ANN) is applied in the prediction of the temperature function by learning the trend between AC energy consumption and indoor temperature. The temperature functions estimate the input temperature, a variable that estimates AC's reward. The ANN architecture consisted of three hidden layers. A layer computes the weighted sum and constant bias parameters. Rectified Linear Unit (ReLu) transfers data from one layer to the next. The Adam optimization algorithm was adopted in training the ANN. The algorithm was tested on a simulation platform.

2.6.4 Demand Response with Reinforcement Learning and Fuzzy Reasoning

The authors [27] proposed a demand response scheme using RL with a single agent and integrates fuzzy reasoning to approximate values for reward functions. Human preference is considered in the control feedback as a state at each time step t.

In their modeling of the home energy management system, appliances are categorized into shiftable (appliances which can be rescheduled on basis of appliance setting and preference setting) and non-shiftable (appliances that cannot be shifted to another time regardless of electricity prices). Q-learning (an off-policy RL technique) was considered in selecting an optimal decision. The MDP constituted state-space with all the possible states in terms of power demand and electricity price signals. The power demand is categorized into low, average, and high while electricity price signals either cheap or expensive, action space provides an agent with three action options based on customer preferences, load priority, real-time price, and power demand. Do-nothing action assumes normal operation and thus no need to shift an appliance, shifting action considers shifting the lowest priority device and valley filling turns on a shiftable appliance with the highest priority and reward function implemented using fuzzy logic which approximates the numerical reward for a certain action and state. The actions with the highest reward values are considered optimal and corresponding actions implemented.

2.6.5 Smart home appliances operational time scheduling

The author [28] proposed real-time scheduling of household appliance operational time based on Q learning. The objective is to minimize energy consumption and ensure the achievement of the comfort level. The problem is to establish an intelligent agent that will optimally schedule the operational time of appliances while considering the dissatisfaction level that arises based on such actions. A home is modeled as an MDP environment, depending on the needs of consumers' changes from one state to another. Set of actions consists of turning on, off, and changing the power level; an agent to design a policy based on historical energy consumption of appliances; reward consisting of a reward vector that is designed to incorporate a human comfort level.

Appliances are categorized into deferrable, non-deferrable, and controllable appliances. Deferrable appliances can be shifted to another time slot or time of the day, Non-deferrable appliances cannot be rescheduled to another time of the day, and controllable appliances can be operated at varied power ratings. An agent is assigned to each appliance and becomes coordinated. The output is an optimal decision of turning on or off of appliances and changing power levels.

2.6.6 Data-driven multi-agent Reinforcement Learning

The author [29] designed a multi-agent RL intending to achieve an efficient home-based DR by modeling a one-hour ahead scheduling of smart appliances for a home energy management system with PV generation. The finite MDP is used to formulate the problem with four agents corresponding to non-shiftable, EV load, time-shiftable, and power shiftable appliances. The MDP framework constituted an agent whose function is to observe the state and select an action. The agent then receives a new state and selects a new action for next time. States included current and future electricity prices, the output of the PV generation. Action is the scheduling of energy consumption for each home appliance. The action is set for each agent. The actions for energy consumption by appliances are determined an hour ahead at each time slot.

The proposed RL approach consists of two parts. The first part is a training of the Extreme Learning Machine (ELM) algorithm which is based on the feed-forward Neural Network. The ELM, using previous 24-hr data, predicts the 24-hr future trend on electricity prices and solar PV generation output. The predicted data is input to the second part which is a Q-learning algorithm designed to make hour-ahead decisions on energy consumption based on optimal policy. The optimal Q value is obtained using the Bellman equation. RL solution can be summarized to entail three algorithms, first algorithm the main function that initializes the parameters of the Q learning. The second algorithm is a feedforward NN with 24-hr data on electricity prices and solar generation as its input. The output is the predicted information on electricity price and solar generation for the next hour. The third algorithm is the Q-learning algorithm that makes scheduling decisions based on optimal policy.

2.6.7 Real-Time Scheduling of Appliances

The authors [30] researched real-time residential demand response to minimize the cost of electricity and maximize user comfort. They presented an optimal scheduling strategy of appliances based on deep reinforcement learning (DRL) considering both discrete and continuous policies. An approximate policy was design based on the neural network (NN) to learn the optimal scheduling strategy from high-dimensional data of real-time pricing, states of an appliance, and outdoor temperature. The NN is trained using a policy search algorithm.

Home appliances were modeled as deferrable, regulated (power adjustable), and critical appliances (don't participate in DR) and problem formulated as an MDP process to be solved using DRL. The MDP structure consists of states as real-time electricity prices, outdoor temperature, and state of all appliances. The operational state for each appliance is a function of three variables – operating status with its value as 1 or 0 if appliances operate or otherwise, a measure of task progress variable and task attribute of an appliance. Actions include binary control action variables/ discrete (deferrable appliances), continuous control variables (regulated appliances). Reward function modeled on three aspects: thermal comfort index, electricity cost, and consumer range anxiety (estimates the consumers fear of the battery's insufficient energy to serve up to certain time). An objective function that finds the optimal policy from a set of all policies, which maximizes the cumulative rewards over time.

In solving the MDP, a neural network-based stochastic policy is adopted to determine the optimal policy. Two functions are used to estimate the optimal policy depending on whether it's a discrete or continuous action. Bernoulli distribution and Gaussian distribution functions are used to estimate the approximate policy when the action is discrete and continuous, respectively. NN policy network was used to determine the parameters for the distribution functions by learning them. The architecture of the NN takes in the input parameters (past electricity prices, outdoor temperatures, and states of all the appliances) and outputs the discrete and continuous actions by Bernoulli and Gaussian distribution functions respectively.

2.7 Research Gaps

Demand response (DR) has been expressed as a reinforcement learning problem formulated using the Markov Decision Process (MDP). The objective function authors considered in their research ranges from minimizing the cost of electricity, optimal scheduling of appliances and efficient DR strategies to the achievement of comfort levels by users.

The MDP formulation scope considered modeling agents as single or multi-agents. A single agent learns the entire environment consisting of all appliances while multi-agent are integrated into each appliance. Appliances are classified as deferrable, power-adjustable and critical appliances. Other classifications considered shiftable, non-shiftable, power-shiftable and Electric Vehicle loads. Rewards were modeled as electricity cost, thermal comfort index for Air-Conditioning systems, value estimates from fuzzy logic control. Action space entailed either binary control, continuous control variables.

Q-learning algorithm is a predominant tool in reinforcement learning that aids in decision making when establishing the optimal policy. However, the research continuity on DR algorithms is still fashioned in simulation environments. Other algorithms are still challenged with a curse of dimensionality problem as they constitute large state-action space which reduces the speed of learning by an agent. Also, multi-agent systems involved assigning an agent to each appliance which seems a complex system for a small residential environment. There is limited input in formulating uncertainties where consumers' preference in curtailing demand is not considered. Also, load classification adopts the rule of thumb or general standards not tailored to the user's load demand. Additional research entailed developing Artificial Neural Networks that introduce complexity in demand response integration. Consumer discomfort considers thermal applications and not dissatisfaction related to other appliances.

This research focused on designing a simplistic model-free and an off-policy algorithm that targets to reduce the curse of dimensionality experienced in large state-action space. Also, quantifying human dissatisfaction or providing estimates may seem erroneous as it is potentially feasible in a simulation environment only. To provide flexibility in demand response, this research included an interactive approach where load demand is grouped into four main categories based on consumer's priority. This priority is dynamic depending on the consumer's energy demand from time to time. The consumer has a preference for load prioritization and decision-making on whether to implement load management strategies such as load shifting or add loads from low priorities when tariffs are deemed low. Consumer feedback was integrated using a knowledge base approach that evaluates a reward comparison between algorithm and consumer action vectors purposely to update fuzzy logic rules. The overarching goal was to maintain user satisfaction through the provision of load management strategy options and knowledge base while minimizing the cost of electricity.

The associated limitation of this research is that a time of use tariff framework is derived from historical electricity tariffs which are statically analyzed to define a point at where tariffs are considered low or high. This results in a definite state-action space that is easier to manage by a single agent. The set-up considers a combination of deterministic and stochastic actions while rewards are deterministic. Reward function proposed estimation of rewards for each action based on a set of base linguistic rules. A fuzzy logic system is applied in reducing uncertainties such as understanding the operation time of an appliance. This is achieved by estimating rewards based on load demand rather than energy to be consumed.

Table 2-2 summarizes the scope of related works and their limitations or gaps flagged from the existing literature.

Table 2-2 Related research work and research gaps

Reference Studies	Environment/ Type	Year	Scope	Limitations/ Gaps/Outcome
[25]	Simulation	2019	The authors modeled a community energy system consisting of an energy pool, intelligent and non-intelligent users. Local energy pool can trade with its users: energy surplus from DGs and smart homes. The objective was to model the trading concept as an MDP and determine an optimal decision (sell or buy, charge, or discharge) using Q learning. The fuzzy Inference system converts the infinite states or models of energy price and the State of Charge to the finite state for a solution by Q-learning. Otherwise infinite states will lead to no optimal solution finding.	The use of Fuzzy logic only approximates the continuous battery and price models. This introduces errors due to approximations based on inference rules. Human feedback is not incorporated into the control system. The author proposes future work to adopt Deep Q-learning and develop a model for a continuous MDP problem.

[26].	Simulation	2020	Real-time DR for a residential consumer using DRL for both discrete and continuous actions. The objective is to minimize the cost of electricity and maximize the user's AC thermal comfort. NN was used to estimate the Q-function using Bernoulli and Gaussian distribution function for discrete and continuous actions, respectively.	The author's approach focuses on consumer comfort for a specific appliance which is thermal comfort. No illustration of how other appliances are integrated into consumer feedback.
[27]	Simulation	2020	Authors modeled residential demand response using RL with a single agent controlling 14 appliances. Human feedback is integrated into the control logic. Three actions suggested load shifting, valley-filling, and no-action. The reward model is a Fuzzy system and approximates the reward for each action. The input to the Fuzzy model is energy demand and electricity prices.	Whereas human feedback is reported to have been included in the control system, still the proposed system causes dissatisfaction to users. The fuzzy model approximates the reward based on a set of fuzzy inference rules. Power demand classified into three levels with no clear benchmark for setting the energy classification

[28]	Simulation	2020	Develop an intelligent agent that will optimally schedule the operational time of appliances and storage to minimize energy consumption and improve user comfort. The problem was formulated as an MDP with Q-learning for the decision-making of an optimal policy. Human comfort is incorporated in the reward function.	Human comfort is integrated into the rewards function which implies that consumer integration of other appliances is missing as an input function.
[29], [30]	Simulation	2020	Design of a multi-agent RL that achieves an efficient home-based DR by scheduling appliances and Electric Vehicles load an hour ahead. RL consists of two phases, the first phase is the training of the feed-forward Neural Network Extreme Learning Machine (ELM) algorithm that predicts a 24hr future trend on electricity prices and solar PV generation. The second phase is the Q-learning algorithm designed for an hour -ahead decision-making based on optimal policy. Real-world electricity prices and solar PV generation data used in training the feed-forward NN.	Based on NN which according to the approach seems subjective based on the interpretation of an hour ahead prediction of electricity prices. Doesn't address the question of what happens when there is a sharp rise in electricity prices. The proposed methodology sounds complex for a simple residential system and likely to introduce delays in learning by an agent.

2.8 Conclusion

This chapter entailed a review of machine learning algorithms particularly reinforcement learning algorithms and their application in managing demand response for residential systems. The review established the Q-learning is an off-policy reinforcement learning algorithm predominantly adopted in implementing demand response applications. The implementation of algorithms was found to be limited to simulation environments which utilities consider as theoretically proven approaches lacking interaction with real loads. Limited deployment of such algorithms in physical systems has made it impossible for other countries to evaluate demand response opportunities on the basis such algorithms perform best only in simulation environments. In terms of the performance of algorithms, the curse of dimensionality was found to be an outstanding gap. Human feedback integration is a critical component that measures the human dissatisfaction levels against the algorithm's optimal policy. Other research works considered human feedback in terms of thermal comfort limited to air conditioning without considering similar human comfort for other appliances. A fuzzy logic rule system was proposed to deal with continuous states or uncertainties by introducing discrete approximates.

Chapter 3 Methodology

3.1 Research Framework

The research objective is a reduction in overall energy cost for a residential environment by voluntary application of load management strategies optimized on a time of use (ToU) tariff which indirectly optimally schedules the operation of appliances. The algorithm targets to modify a residential load curve depending on a ToU tariff, login consumer feedback and update fuzzy rules associated with the dissatisfaction. This framework proposes a Q-learning algorithm to achieve the research objective. The proposed framework consists of

- Problem formulation - The objective is formulated as a reinforcement learning problem via a Markov Decision Process (MDP). The MDP consists mainly of a single agent, environment and fuzzy logic controller; Appliance Classification - appliances will be classified into two main groups, non-schedulable and schedulable appliances. Schedulable appliances are further classified into interruptible and non-interruptible.
- Knowledge database – consists of data on total demand and electricity tariff which is statically analyzed to derive updates to the fuzzy rules. It is anticipated that a change in total demand due to priority preference by consumers impacts load classification and subsequently the reward structure.
- Matlab Graphical User Interface (GUI) – consists of controls that facilitate the algorithm's behavior analysis on changes to load and other factors. The GUI can also integrate visual analysis on the performance of the algorithm and accepts feedback from the consumer.
- Testing Environment includes physical and simulation testing environments. Matlab is interfaced with Arduino through serial communication.

The design architecture illustrating the Integration of the Smart Home Energy Management System (SHEMS) with the MDP problem formulation is given in Figure 3-1. A detailed breakdown of the structure is given in the testing architecture described in the subsequent sub-section. The SHEMS receives the electricity tariffs from the grid through the smart meter and load demand from the user. The SHEMS houses the algorithm which outputs instructions in form of optimal action policy and communicates to the power control unit. Appliances have current sensors that detect the operation status of appliances and communicate to the algorithm in binary one and zero forms.

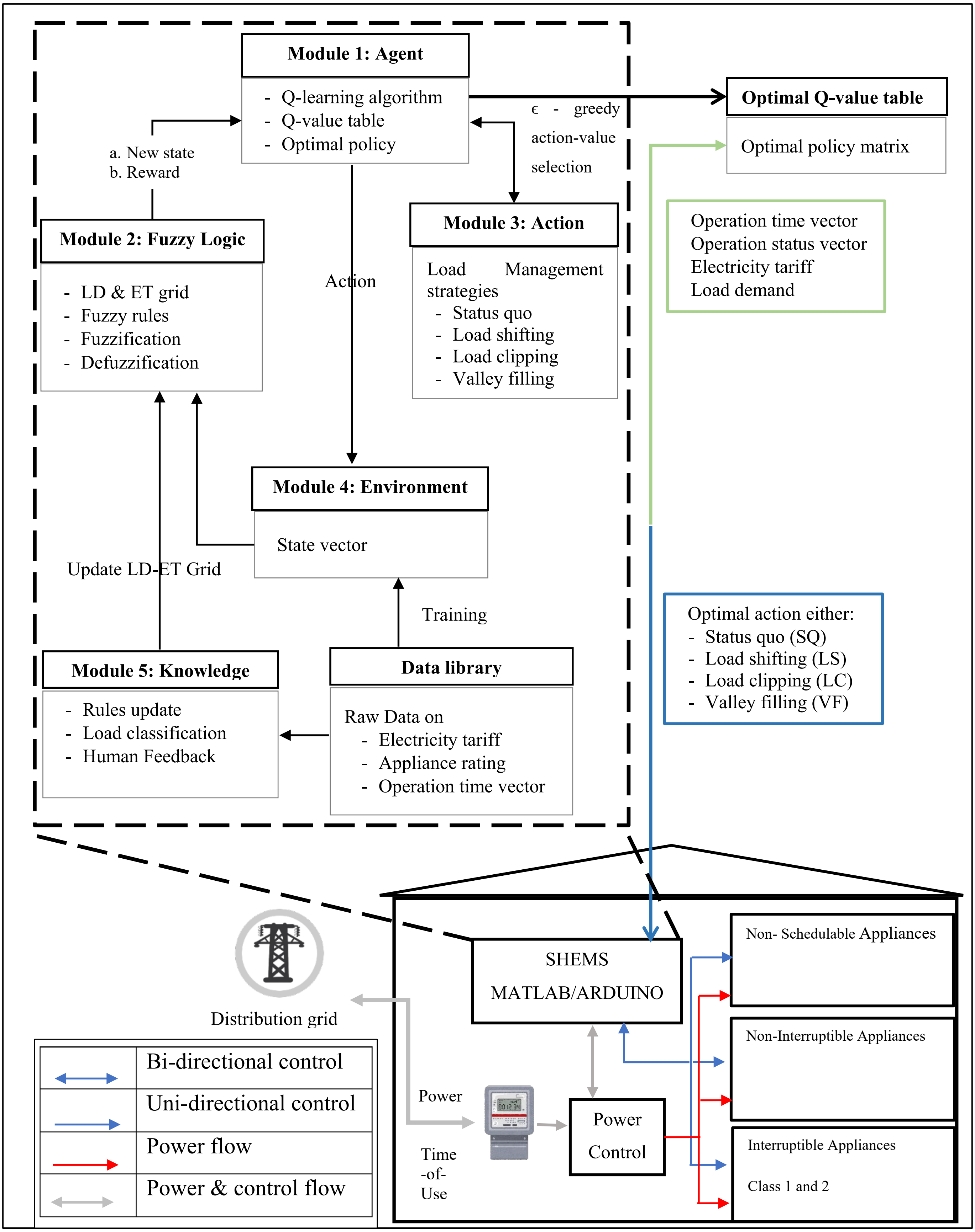


Figure 3-1 Designed architecture of the Smart Energy Management System

3.2 Markov Decision Process (MDP) Model

3.2.1 Environment

3.2.1.1 Load classification

The environment consists of home appliances classified into two main categories mainly non-schedulable and schedulable. Non-schedulable appliances are considered mandatory and therefore excluded in the demand response participation. However, understanding their contribution to load demand is critical in establishing an optimal policy. Schedulable (Interruptible and non-interruptible) appliances are primary participants in demand response and provide the leverage to deploy load management strategies per the time-of-use tariff to realize energy savings. Load classification and load control level emanates from arranging load demand for the appliances according to consumer preference and priority and computing their cumulative load demand, respectively.

The total demand, P_t from all the appliances at any given time is given by equation

$$P_t = P_t^{NS} + P_t^{NI} + P_t^{IN1} + P_t^{IN2} \tag{3.1}$$

The respective load demand for the appliances is computed as an elementwise product of load demand and operation status vectors.

$$P_t^{NS} = P_{rat}^{NS} .* H_t^{NS} \tag{3.2}$$

$$P_t^{NI} = P_{rat}^{NI} .* H_t^{NI} \tag{3.3}$$

$$P_t^{NI1} = P_{rat}^{IN1} .* H_t^{IN1} \tag{3.4}$$

$$P_t^{NI2} = P_{rat}^{IN2} .* H_t^{IN2} \tag{3.5}$$

The parameters P_{rat} represents the appliance rating vector, H_t are the appliance operation status vector in a Boolean form with 0 and 1 indicating the appliance is off (false) and on (true), respectively.

Load control levels are defined cumulatively by adopting load demand for each appliance category.

$$LCL_1 = P^{NS} \tag{3.6}$$

$$LCL_2 = P^{NS} + P^{NI} \tag{3.7}$$

$$LCL_3 = P^{NS} + P^{NI} + P^{IN1} \tag{3.8}$$

$$LCL_4 = P^{NS} + P^{NI} + P^{IN1} + P^{IN2} \tag{3.9}$$

Where P^{NS}, P^{NI}, P^{IN1} and P^{IN2} is the total load demand for non-schedulable, non-interruptible, priority one and two interruptible appliances, respectively.

Let the static Time of Use (ToU) tariff signal at time *t* be given by $\gamma(t)$, then the cost of energy in each appliance category is given by the following equations.

$$\theta_{NS} = \sum_{t=1}^{24} {P_{rat}}^{NS}.H^{NS}(t).\gamma(t) \tag{3.10}$$

$$\theta_{NI} = \sum_{t=1}^{24} {P_{rat}}^{NI}.H^{NI}(t).\gamma(t) \tag{3.11}$$

$$\theta_{IN1} = \sum_{t=1}^{24} {P_{rat}}^{IN1}.H^{IN1}(t).\gamma(t) \tag{3.12}$$

$$\theta_{IN2} = \sum_{t=1}^{24} {P_{rat}}^{IN2}.H^{IN2}(t).\gamma(t) \tag{3.13}$$

3.2.1.2Objective Function

The goal that the algorithm needs to achieve is minimizing the cost of electricity. The total cost of energy consumption by appliances in the four categories is given by

$$\theta_{total\ cost} = \theta_{NS} + \theta_{NI} + \theta_{IN1} + \theta_{IN2} \tag{3.14}$$

The objective function targets to minimize the aggregated cost of electricity due to energy consumption by all the appliances and can be formulated as :

$$\boldsymbol{Minimize}\ (\theta_{total\ cost}) = \theta_{NS} + \theta_{NI} + \theta_{IN1} + \theta_{IN2} \tag{3.15}$$

3.2.2 Agent

A single agent was trained using data from a residential consumer and learns on policies based on the data (environment). The agent's key objective is to establish an optimal policy from a given set of load management strategies. The agent triggers an action per an action-value selection policy and receives a new state and reward for its action. It updates a policy Q-value table for several iterations until convergence is achieved.

3.2.3 State Space

The set of state-space consists of the total demand for appliances and electricity tariff at time t. Load demand for appliances is categorized into four distinct load control levels depending on the priority and category of the appliance. The electricity tariff Time of Use (ToU) limit assumption is derived from the statistical analysis of historical domestic-ordinary electricity cost data since November 2018 charged by Kenya Power and Lighting Company [31]. The data set statistical properties are a maximum, minimum, mean and standard deviation of US$ 0.1533, 0.1377, 0.1438 and 0.0052 per kWh, respectively. The dataset has 60.47 % of data above the mean value while the rest below it. This research categorized electricity tariffs into two main groups low (below the mean) and high tariff (above the mean).

3.2.4 Action Space

The action space consists of a set of four actions corresponding to three load management strategies (load clipping, valley filling and load shifting) and status quo (no action). An action is either Highly Recommended (HR), Recommended (R), Least Recommended (LR) or Not Recommended (NR).

- Load clipping strategy is HR when the consumer exceeds load control three (LC 3) for both low and high ToU electricity prices. It involves switching off appliances with priority assignment four (4). All the other actions are either LR or NR.
- Valley filling is HR during low or moderate demand and low ToU electricity prices to optimize the low tariff window. It involves filling of appliances with priority assignment three (3) or four (4). It is NR during high electricity tariffs and loads above load control two (LC2).
- Load shifting is HR when the load demand is above LC 2 and when the tariff is high and additional load would result in electricity cost. It is NR in other scenarios. This implies deferring appliances with priority assignments 3 and 4 until a valley filling window opens.
- Status quo implies that no action should be taken when operating below LC 3 and at a low tariff. Additionally, when the load demand is moderate and prices are high, the consumer may opt not to take action apart from load shifting which is LR.

The balance in the action space is given in Figure 3-2. Load shifting and clipping actions are compensated by valley-filling. The action of load shifting and clipping is anticipated to cause energy savings while valley filling will increase the initial cost. The goal of the algorithm is to ensure the net costs before and after demand response is positive implying energy savings for a particular demand response regime.

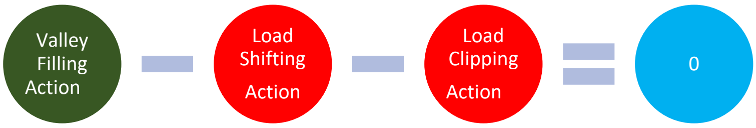


Figure 3-2 Balance in action space

The action space per the load demand and ToU electricity price grid Figure 3-4 is assigned a weight which is essential when integrating feedback from consumers as described in section 3.5. The weights are shown in Table 3-1.

Table 3-1 Weights of recommendation for action-space

Action Recommendation	**Weight, υ**
Highly Recommended (HR)	0.4
Recommended (R)	0.3
Least Recommended (LR)	0.2
Not Recommended (NR)	0.1
Total	**1.0**

3.2.5 Reward Function

The objective function consists of an approximate numerical reward estimated by fuzzy logic systems. A Fuzzy logic system constitutes a crisp input (the load demand and ToU electricity tariff) and crisp output is the numerical reward approximated by the system’s inference engine.

3.3 Fuzzy Logic Systems

3.3.1 Introduction to Fuzzy Logic Systems

A fuzzy logic system is a mathematical knowledge or rule-based system developed from a set of linguistic rules and transforms this knowledge into a non-linear mapping of inputs which can be numbers or vectors of numbers to outputs. It is broadly applied in a highly complex system characterized by uncertainty in behavior and where a fast approximate solution is needed. According to [32], the three main types of fuzzy logic systems commonly used include (i) fuzzifier and defuzzifier, (ii) pure and (iii) Takagi-Segeno-Kang (TSK) fuzzy systems. A fuzzy system with fuzzifier and defuzzifier is commonly used as it eliminates problems associated with pure and TSK fuzzy systems. Figure 3-3 shows the basic configuration of a fuzzy logic system with fuzzifier and defuzzifier consisting of four key elements with descriptions and mathematical definitions explained in the subsequent subsections.

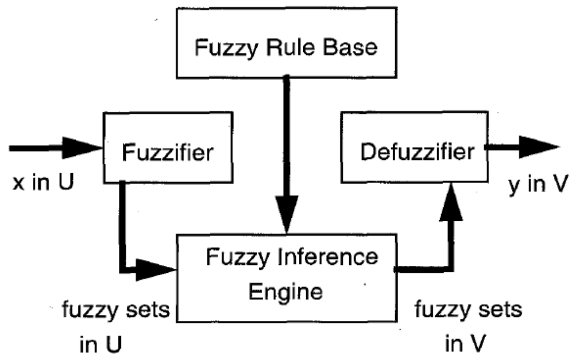


Figure 3-3 Basic configuration of a fuzzy system with fuzzifier and defuzzifier

A fuzzifier performs fuzzification process which is the conversion of crisp input into a fuzzy set. A fuzzy set constitutes elements in a vector space and a set of membership functions that maps the vector space onto [0, 1]. A fuzzy rule base is the heart of a fuzzy system consisting of a collection of fuzzy IF-THEN rules. These rules are characterized by membership functions. A fuzzy inference engine (algorithm) combines fuzzy rules into a mapping of the input fuzzy set space to output a fuzzy set base on a fuzzy logic control principle. A defuzzifier performs defuzzification which is a conversion of fuzzy sets into crisp quantities for additional processing.

3.3.2 Fuzzy Sets and Basic Operations on Fuzzy Sets

A fuzzy set A in a universe of discourse U can mathematically be represented as a set of ordered pair consisting of an element x and its membership value $\mu_A(x)$ as [32],

$$A = \left\{ \left(x, \mu_A(X) \right) | x \epsilon U \right\} \tag{3.16}$$

The membership function of a fuzzy set is given as a continuous function within the interval [0, 1]. The basic operations on fuzzy sets, say A and B in the same universe of discourse U include union, intersection and complement operations.

A union of A and B in a fuzzy set $A \cup B$ in U with a membership function defined as

$$\mu_{A \cup B}(x) = max\left[\mu_A(x), \mu_B(x) \right] \tag{3.17}$$

An intersection of A and B is a fuzzy set $A \cap B$ in U with a membership function defined as

$$\mu_{A \cap B}(x) = min\left[\mu_A(x), \mu_B(x) \right] \tag{3.18}$$

A compliment of A is a fuzzy set $\bar{A}$ in U with a membership function defines as

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \tag{3.19}$$

Fuzzy sets characterize linguistic variables defined in their universe of discourse in which the variable is defined. Linguistic variables assume words in natural language as their value. In the context of fuzzy theory, a linguistic variable consists of a vector (X, T, U, M) where X is the linguistic variable name, T is a set of values that X can assume in linguistic form, U is the physical domain the variable X assumes in quantitative values and M is a rule that relates each linguistic value in T with a fuzzy set in U.

3.3.3 Fuzzy Rule Base and Fuzzy Inference Engine

A fuzzy rule base as the heart of the fuzzy system constitutes a set of IF-THEN rules.

Let ${A_i}^{(k)}$ and ${B_i}^{(k)}$ be fuzzy sets in $U_i \in R$ and $V_i \in R$, respectively, $x = (x_{1,} x_2, .., x_n)^T \in U$ and $y \in V$ be inputs and outputs variables of the fuzzy system, the rules can be defined in the format [32],

$$Ru^{(k)}: IF\ x_1\ is\ {A_1}^k\ and\ \ldots\ and\ x_n\ is\ {A_1}^k\ , THEN\ y\ is\ B^k \tag{3.20}$$

A collection of other fuzzy rules from $k = 1, 2, \dots, Y$ where Y is the number of rules in the fuzzy rule-based is referred to as canonical form IF-THEN rules. This concept is extended to the load demand-electricity price grid which is self-developed as Figure 3-4. From the figure, eight rules in a canonical form are defined in the self-developed Table 3-2.

The fuzzy set A includes the load controls in a universe of discourse V[Minimum, Maximum electricity prices] equivalent to Load control 4 (LC4) and status of electricity prices (whether Low or High) in a universe of discourse derived from historical tariff data.

Fuzzy set B constitutes action space linguistic form Highly Recommended (HR), Recommended (R), Least Recommended (LR) and Not Recommended (NR).

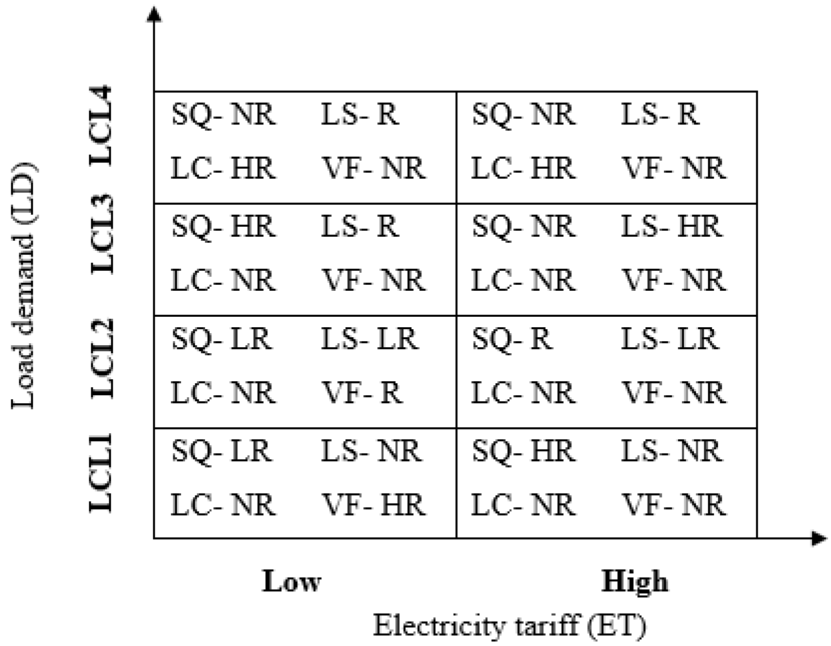


Figure 3-4 Load demand/electricity tariff grid

The fuzzy inference engine uses the fuzzy logic principles to combine the IF-THEN rules defined in the fuzzy rule base into a mapping defined as

$$fuzzy\ set\ A^k in\ U \rightarrow fuzzy\ set\ B^k\ in\ V \tag{3.21}$$

Table 3-2 Canonical form of the rule base

Rule 1	If (LD is LCL1) and (ET is LP) then (SQ is NR)(LS is NR)(LC is NR)(VF is HR) (1)
Rule 2	If (LD is LCL1) and (ET is HP) then (SQ is HR)(LS is NR)(LC is NR)(VF is NR) (1)
Rule 3	If (LD is LCL2) and (ET is LP) then (SQ is LR)(LS is LR)(LC is NR)(VF is HR) (1)
Rule 4	If (LD is LCL2) and (ET is HP) then (SQ is HR)(LS is LR)(LC is NR)(VF is NR) (1)
Rule 5	If (LD is LCL3) and (ET is LP) then (SQ is HR)(LS is R)(LC is NR)(VF is NR) (1)
Rule 6	If (LD is LCL3) and (ET is HP) then (SQ is NR)(LS is HR)(LC is NR)(VF is NR) (1)
Rule 7	If (LD is LCL4) and (ET is LP) then (SQ is NR)(LS is NR)(LC is HR)(VF is NR) (1)
Rule 8	If (LD is LCL4) and (ET is HP) then (SQ is NR)(LS is NR)(LC is HR)(VF is NR) (1)

The fuzzy rule base often constitutes more than one rule hence needs to infer with the rules. Two methods to infer with a set of rules include composition and individual-rule-based inference [32]. The composition-based inference system combines all the fuzzy rule base into a single fuzzy so that it's viewed as a single IF-THEN rule. Mamdani inference method, a type of composition-based inference, is adopted based on intuitive appeal. Mamdani combination is defined as a single fuzzy relation Q_M,

$$Q_M = \bigcup_{k=1}^{Y} Ru^k \tag{3.22}$$

Where $Ru^{(k)}$ is a fuzzy relation in UxV representing the fuzzy rule in (3.20)

Two types of inference engines that are commonly used in fuzzy systems control are the Product and Minimum Inference Engine. They have an advantage in terms of computational simplicity. A minimum inference engine is adopted in this research defined as,

$$\mu_{B^k}(y) = \max_{k=1}^{Y} [\sup_{x \in U} min\left(\mu_{A^k}(x), \mu_{A_1{}^k}(x_1), \ldots, \mu_{A_n{}^k}(x_n), \mu_{B^K}(y)\right)] \tag{3.23}$$

3.3.4 Fuzzifiers

A fuzzifier refers to the mapping of a real value $\boldsymbol{x}^{\,max} \in \boldsymbol{U} \subset \boldsymbol{R}^n$ to a fuzzy set $A^k\ in\ U$ [32]. The criteria for designing a fuzzifier require that a fuzzifier should simplify the fuzzy inference engine computations and suppress the noise in case the input is corrupted by noise. Fuzzifiers can be classified into Singleton, Gaussian and triangular fuzzifier with preference given to the triangular method in this research.

Singleton fuzzifier maps a real value $\boldsymbol{x^{max}} \in \boldsymbol{U}$ to a fuzzy singleton $A^k\ in\ U$ characterized by a membership value of 0 or 1.

$$\mu_A(x) = \begin{cases} 1\ if\ x = x^{max} \\ 0\ if\ otherwise \end{cases} \tag{3.24}$$

Gaussian fuzzifier maps a real value $\boldsymbol{x^{max}} \in \boldsymbol{U}$ to a fuzzy set $A^k\ in\ U$ characterized by a Gaussian membership function,

$$\mu_A = e^{-\left(\frac{x_1 - x_1{}^{max}}{a_1}\right)^2} \cdots e^{-\left(\frac{x_n - x_n{}^{max}}{a_n}\right)^2} \tag{3.25}$$

Triangular fuzzifier maps a real value $\boldsymbol{x^{max}} \in \boldsymbol{U}$ to a fuzzy set $A^k\ in\ U$ characterized by a triangular membership function,

$$\mu_A(x) = \begin{cases} \left(1 - \frac{|x_1 - x_{max}|}{b_1}\right) \cdots \left(1 - \frac{|x_n - x_{max}|}{b_n}\right) & if\ \left|x_i - x_{i,max}\right| \le b_i, i = 1,2,\ldots,n \\ 0 & if\ otherwise \end{cases} \tag{3.26}$$

3.3.5 Defuzzifiers

Defuzzifier maps the output of the inference fuzzy engine to a crisp point. A fuzzy set B^k in $V \complement R$ is mapped to a crisp point $y^{max} \in V$ [32]. Based on the criteria for plausibility, computational simplicity and continuity, this research adopts the center of gravity (CoG) defuzzifier.

The CoG defuzzifier specifies y^*as the area center covered a membership of B^k as

$$y^* = \frac{\int_V y\left(\mu_{B^k}\right)(y)dy}{\int_V (\mu_{B^k})(y)dy} \tag{3.27}$$

The defuzzifier in this case outputs the approximate reward based on the crisp input (load demand and electricity prices).

3.4 Q Learning Algorithm

3.4.1 Introduction to Q-learning algorithm

Q-learning algorithm is a temporal difference learning algorithm and an off-policy reinforcement learning that aims to learn optimal policy from experience without a clear model of the environment. The algorithm approximates the current optimal action-value q_* using the Bellman equation,

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_t + \gamma max_a Q\left(s_{t+1}, a\right) - Q(s_t, a_t)\right] \tag{3.28}$$

Given a set of all states defined by $S \in S^+$ where S^+ is a set all states plus terminal state, Q-learning algorithm aims to compute the value of taking an action a in state s and determines the optimal policy, $q_*(s, a)$ from a set of actions $a \in A(s)$for that particular state. A policy defines the steps an algorithm takes to make a decision. The parameters α and γ represent the learning rate of the algorithm and discount factor [24], [33], respectively. The set of states for this research can be derived from the load demand-action grid in Figure 3-4 while actions $a \in A(s)$ from the action space defined in section 3.2.3. The resulting set of eight states is given in Table 3-3

Table 3-3 Set of states depending on load demand and tariff

	Load demand			
Electricity Tariff	LCL1	LCL2	LCL3	LCL4
LP	LCL1, LP	LCL2, LP	LCL3, LP	LCL4, LP
HP	LCL1,HP	LCL2,HP	LCL3,HP	LCL4,HP

A potential way for the classification of reinforcement learning algorithms is by how policies are improved in the process of learning. On-policy algorithms apart from learning from its self-generated data, the policy acting on the environment is the same as one that improves learning. Off-policy algorithms consist of a two-policy combination which are behavior and target policies. Behavior policy interacts with the environment to collect information about the environment while a target policy learns and improves.

3.4.2 Exploration and Exploitation

In selecting which actions the agent needs to take, one of the key challenges in reinforcement learning is the tradeoff between exploitation and exploration by an agent. The exploitation-exploration dilemma introduces two options, whether an agent should exploit what is already learned and known or explore new options. Polices can be deterministic or stochastic.

Deterministic policies limit the understanding of the knowledge about the environment by considering a limited set of states and what has already been learned. Stochastic policy on the other end explores the environment to identify new options and establish better policies. A balance between exploitation and exploitation ensures the environment is explored whilst taking into consideration previously learned knowledge.

There are several exploration techniques such as include $\varepsilon - greedy$ and Boltzmann exploration. Boltzmann exploration is more accurate and complicated.

This research adopts the $\varepsilon - greedy$ exploration technique which ensures actions are selected randomly (exploration) and greedily (exploitation) with a probability ε and $1 - \varepsilon$. For example, ε is 0.2, which implies twenty percent of actions will be selected randomly. It will also be considered to decrease over time at later stages. The agent at this phase has obtained confidence in the knowledge thus needs to avoid excessive exploration.

$$a_t = \begin{cases} random\ action & with\ probability\ \varepsilon \\ maxQ_t(a) & with\ probability\ 1 - \varepsilon \end{cases} \tag{3.29}$$

3.4.3 Returns and Episodes

One of the primary goals of an agent is to maximize cumulative rewards in a particular time slot. Denote sequence of rewards as $R_{t+1}, R_{t+2}, R_{t+3}, \ldots$ so that the expected return is maximized. The maximized return is considered a function of the sum of all rewards.

$$G_t = R_{t+1} + R_{t+2} + R_{t+3} + \cdots + R_T \tag{3.30}$$

Where T is the final time step or episode.

Sometimes the current state is naturally broken into subsequences called episodes which ends when a terminal state is reached. After a terminal state, the environment is reset to the next starting state. Tasks with episodes are called episodic tasks characterized by a set of non-terminal states, set of all sets including non-terminal state and termination T. Other agent-environment interaction may constitute continuity tasks where episodes are continuous and the final time step $= \infty$, therefore, an infinite return. The discounting method is applied so that an agent chooses actions to objectively maximize the sum of discounted rewards it receives in the future. The discounted return is estimated [34],

$$G_t = R_{t+1,}\gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{3.31}$$

Where γ is the discount rate: $0 \leq \gamma \leq 1$.

3.5 Consumer Satisfaction Knowledge Base

3.5.1 Introduction to Knowledge Base

Consumer feedback is analyzed in the knowledge base which frequently updates the load-electricity price grid given in Figure 3-4 depending on the maximum reward. The algorithm and consumer action vectors are compared to determine consumer dissatisfaction. Binary representation is essential in this sub-section. An action vector consists of binary one and zero digits which correspond to the load management strategies in the order listed in Table 3-4.

Table 3-4 Binary one and zero action vector

SQ	LC	LS	VF	Description
1	0	0	0	Status Quo is the optimal action
0	0	1	0	Load Shifting is the optimal action
0	1	0	0	Load Clippings is the optimal action
0	0	0	1	Valley Filling is the optimal action

3.5.2 Consumer Dissatisfaction

Two action vectors need to be compared during each execution. Consumer feedback is integrated through a variety of options which may include, allowing the consumer to categorize loads depending on the preference and obtaining consumer feedback on the preferred action by the algorithm.

Action vectors by the algorithm and consumer at time t are represented as $W_{Alg,t}$ and $W_{Cons,t}$, respectively. Then the magnitude of the vector can be used to determine consumer dissatisfaction. Consider the length of the action vector difference,

$$\Delta W = \| W_{Alg,t} - W_{Cons,t} \| \tag{3.32}$$

When, $\Delta W = 0$, the consumer is satisfied with the decision of the algorithm. However, the consumer shows dissatisfaction when $\Delta W > 0$. Consumer dissatisfaction with the algorithm's decision is handled by updating the fuzzy rules. However, the reward difference between consumer dissatisfaction and algorithm is maximized.

3.5.3 Update to Fuzzy Rules

The weighting method is used to assign the weights of the linguistic action space in Fuzzy set B as Highly Recommended (HR) – 0.4, Recommended (R) – 0.3, Least Recommended (LR)- 0.2 and Not Recommended (NR) - 0.1. At $time\ t$, the algorithm's action vector and corresponding index are given as:

$$\left[\beta_{alg,t}, \lambda_{alg,t}\right] = max\left(W_{Alg,t}\right) \tag{3.33}$$

Consumer feedback is represented as

$$\left[\beta_{cons,t}, \lambda_{cons,t}\right] = max\left(W_{cons,t}\right) \tag{3.34}$$

The reward comparison is evaluated to maximize the reward, minimize cost and update the fuzzy rule. The difference is reward needs to be greater than zero to guarantee an update to the rules. When $R_t\left(\lambda_{cons,t}\right) - R_t\left(\lambda_{alg,t}\right) > 0$, then the rules are updated depending on the weightage. The load demand and electricity price grid G is assigned the maximum of weighted vector υ.

$$G\left(\left(ET_{n=1\ to\ 2}, LD_{m=1,\ldots,4}\right), \lambda_{alg,t}\right) = G\left(\left(ET_{n=1\ to\ 2}, LD_{m=1,\ldots,4}\right), \lambda_{cons,t}\right) \tag{3.35}$$

$$G\left(\left(ET_{n=1\ to\ 2}, LD_{m=1,\ldots,4}\right), \lambda_{cons,t}\right) = \max(\upsilon) \tag{3.36}$$

The last factor to check is whether the predicted cost of energy due to the update of the fuzzy rules is minimized otherwise, the algorithm will lose track of its objective.

An example of the fuzzy rule update is illustrated in Figure 3-5 should a consumer decide to valley fill his load curve during low tariff while operating at Load Control Level 3. The energy costs based on the proposed update are computed and compared with the initial cost. If the update reveals increased energy cost, the algorithm retains its previous optimal policy and fails to update the rules.

SQ- 0.1	LS- 0.3	SQ- 0.1	LS- 0.3
LC- 0.4	VF- 0.1	LC- 0.4	VF- 0.1
SQ- 0.4	LS- 0.3	SQ- 0.1	LS- 0.4
LC- 0.1	VF- 0.1	LC- 0.1	VF- 0.1
SQ- 0.2	LS- 0.2	SQ- 0.3	LS- 0.2
LC- 0.1	VF- 0.3	LC- 0.1	VF- 0.1
SQ- 0.2	LS- 0.1	SQ- 0.4	LS- 0.1
LC- 0.1	VF- 0.4	LC- 0.1	VF- 0.1

SQ- 0.1	LS- 0.3	SQ- 0.1	LS- 0.3
LC- 0.4	VF- 0.1	LC- 0.4	VF- 0.1
SQ- **0.3**	LS- **0.1**	SQ- 0.1	LS- 0.4
LC- 0.1	VF- **0.4**	LC- 0.1	VF- 0.1
SQ- 0.2	LS- 0.2	SQ- 0.3	LS- 0.2
LC- 0.1	VF- 0.3	LC- 0.1	VF- 0.1
SQ- 0.2	LS- 0.1	SQ- 0.4	LS- 0.1
LC- 0.1	VF- 0.4	LC- 0.1	VF- 0.1

Figure 3-5 Fuzzy rule update using the knowledge base

3.6 Time of Use Tariff

3.6.1 Time of Use Structure

Currently, Kenya adopts a discounted Time of Use (ToU) tariff structure at the commercial and industrial consumer level. Under the discounted ToU regime, consumers are given a fifty percent discount on energy consumption exceeding the set consumption threshold. While the plan is yet to be cascaded to the domestic category, one of the recommendations of the updated least cost development plan [35] is strengthening of ToU in wider scope to strategically ensure that excess generation is utilized during off-peak hours. It is therefore apparent that the recommendation observes demand-side management as a tool and initiative of promoting growth in energy consumption to curb the increasing energy supply through the deployment of ToU.

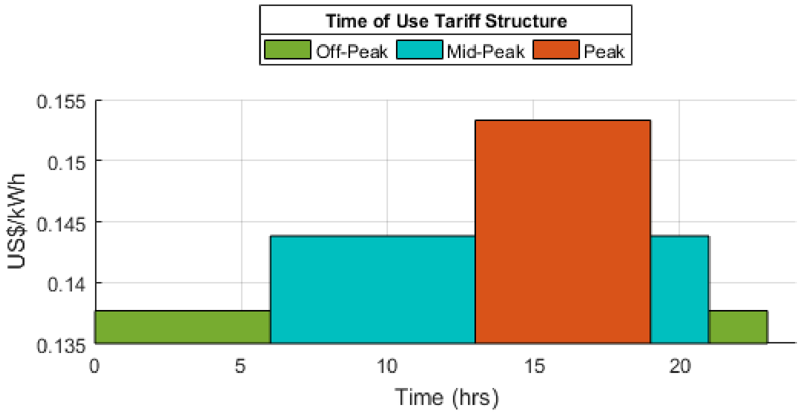


Figure 3-6 Example of Time of Use (ToU) tariff structure

Several structures for the ToU tariff exist with the most common being one with three major tariff segments. These segments include peak, shoulder and off-peak [36]; other countries and authors refer to them as peak, partial-peak and off-peak segments [37].

3.6.2 Static Time of Use Tariffs

Several utilities around the world have implemented a Time of Use plan. An approach to establishing an estimate of ToU pricing is to understand practices in other countries and benchmarking trends that seem reasonable to Kenya's historical electricity prices. Table 3-5 below summarizes the sampled ToU rates (US$/kWh) from various utilities or retailers.

Table 3-5 ToU rates in various countries

Country	**Retailer**	**Off-peak**	**Mid**	**Peak**	**Ratio (Peak/off-peak)**	**Ratio (Mid/off-peak)**
Canada	Waterloo Hydro Inc [38]	0.080	0.119	0.176	2.200	1.488
Australia	AGL Energy Limited [39]	0.170	0.317	0.317	1.865	1.000
Australia	Energy Austria [40]	0.170		0.540	3.176	0.000
Ireland	Electricity Supply Board [41]	0.100	0.140	0.380	3.800	1.400
Sri-Lanka	Ceylon Electricity Board [42]	0.130	0.250	0.540	4.154	1.923
			Min	1.865	Max	1.453

The tariff structure is given in Figure 3-6. Historical tariff data for residential consumers in Kenya are distributed around the mean which is also the shoulder or mid-peak. The product of the minimum ratio of peak to off-peak and shoulder price gives the peak pricing while the quotient of shoulder price and mean ratio of shoulder to off-peak prices gives the off-peak pricing as shown in Table 3-6.

Table 3-6 Time of Use (ToU) plan

ToU Segment	Time of Use (hh:mm:ss)	Tariff (US$/kWh)
Off-peak	00:00:00 – 06:59:59, 22:00:00 – 00:00:00	0.0990
Mid-Peak	07:00:00-13:59:59, 20:00:00-21:59:59	0.1438
Peak	14:00:00-19:59:59	0.2682

3.7 Algorithm Design

In the context of this research and based on the design architecture presented in Figure 3-1, the algorithm integrated the Q-learning algorithm with fuzzy logic systems. A fuzzy logic system has crisp input as the state $S \in S^{+}$ consisting load demand and electricity price with a crisp output as a numerical reward associated with each action in the set of four actions. The key deliverable of the algorithm is an optimal policy table mapping optimal q_{*} which represents a set of optimal actions (policies) that maximize rewards. The algorithm consisted of five major components.

The main component in Table 3-7 integrates all other functions of the algorithm. It defines the environment and other parameters (learning rate of the agent, the discount factor). It calls the Q-learning algorithm and a fuzzy logic system. The Q-learning algorithm in

Table 3-8 is the decision-making component whose objective is to generate a set of optimal policies for the corresponding states. The terminal state is achieved when the number of times the next state is randomly selected is greater than a hundred. The Q-learning algorithm is executed at each time slot of one hour and optimal policy is generated for each slot.

A fuzzy logic system in Table 3-9 accepts the environment parameters and performs conversion into reward estimates for the current state and action. The testing component in

Table 3-10 confirms receipt of optimal policy and performs scenario testing with four bulbs which represent the loads in the four categories. The knowledge base updates the fuzzy rules to match a reward that is satisfactory to the consumer and its impact on the reward due to automatic action. Figure 3-7 is a flowchart of the designed Q-learning algorithm from the determination of the optimal policy to testing of the policy.

Table 3-7 Integrated Q-learning and Fuzzy logic system

A. Main Program
1. Define environment (load demand and electricity tariffs)
2. Define constant parameters such as learning rate $\alpha(0,1]$, discount factor $\gamma \in (0,1]$
3. **For** time t=1 to 24 hours.
4. Execute the **Q-learning algorithm** for decision making
5. **End** for loop
6. Execute Testing

Table 3-8 Q-learning Algorithm

B. Q-1earning algorithm
1. Initialize matrix $Q(s, a)$ for each state-action pair
2. **For** T episodes
3. Initialize state and terminal-state
4. **Do**
5. Use $\varepsilon - greedy$ policy to select an action a_t at current state
6. Execute the fuzzy logic system to obtain a reward, observe current reward and $r(s_t, a_t)$ and randomly selected the next state s_{t+1}
7. Update the Q-value table using (3.28).
8. **End do loop when** a terminal state is achieved
9. **End** the for loop when episode T is reached
10 Output optimal Q-value table

Table 3-9 Fuzzy logic system

C. Fuzzy logic system
1. Fuzzifier load demand and electricity tariff using triangular fuzzifier
2. Use MINIMUM fuzzy inference engine and MAMDANI inference method to map the fuzzy set.
3. Use the center of gravity method to defuzzify the fuzzy set into crisp quantities (**reward**) for additional analysis.

Table 3-10 Testing of optimal policy in Arduino board

D. Testing on the Arduino platform
1. Receive optimal policy
2. Generate scenarios for testing the policy.
3. Turn relay on or off depending on optimal policy

Table 3-11 Update of Fuzzy Rules

E. Knowledge Base
1. The consumer receives algorithm action vector
2. Consumer demonstrates satisfaction or dissatisfaction
3. Compare rewards, predicted energy cost and update the fuzzy rules

3.8 Algorithm Simulation and Testing Environment

3.8.1 Simulation Environment

The algorithm was developed and simulated in a MATLAB & Simulink R2016a environment. The system settings pre-installed with the simulation environment is an Intel (R) Core i7-4500U Central Processing Unit, installed memory of 12 Gigabytes and 2.4 Gigahertz speed.

3.8.2 Testing Environment

3.8.2.1Software Environment

The software environment consists of Matlab and Arduino computer programs divided into sub-functions which depict the architecture in Figure 3-1. Matlab environment consists of four key modules. Module 1 is the agent and the primary module with sub-functions that interact with other modules. Module 2 is a fuzzy reward system that responds to the agent with the reward vector corresponding to the four load management strategies. Module 3 represents the $\epsilon - greedy$ policy action-value selection function that returns an action based on exploration or exploitation. Module 4 models the appliances as the environment. Information on appliance rating, operational status and time belong to this module. Part of this data is used in training the agent purposely for it to develop an optimal policy table.

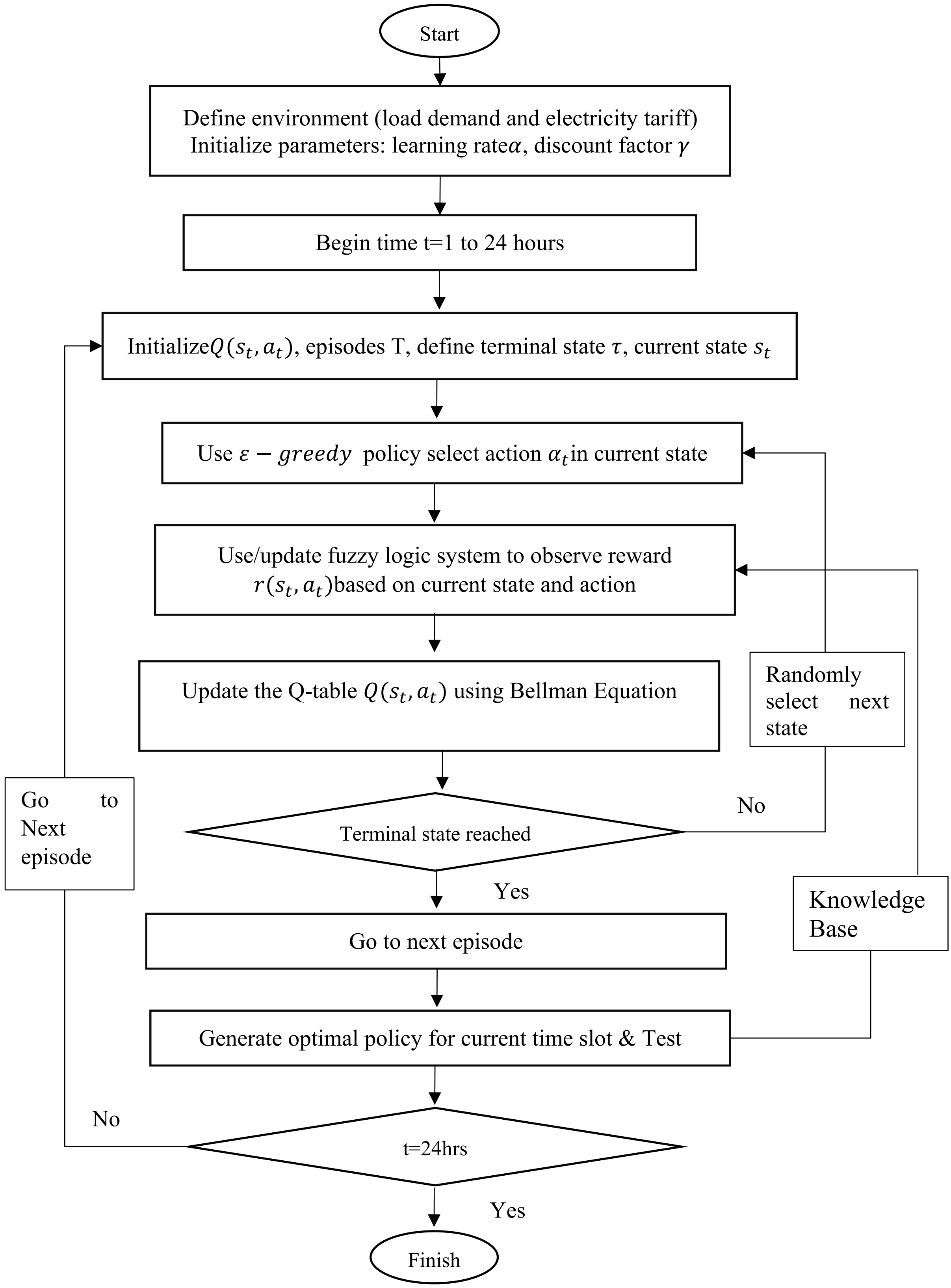


Figure 3-7 Flowchart of the designed demand response algorithm

Arduino environment consists of the implementation code uploaded to its memory.

The Arduino and Matlab environment communicate via serial communication. The optimal policy by the Matlab agent is received serially by Arduino code. The code evaluates the instructions and activates a loop corresponding to the action. Module 5 is the knowledge base that updates the fuzzy rules based on evaluation of consumer feedback on the algorithm optimal policy.

3.8.2.2Hardware Environment

The hardware environment consists of an Arduino Uno R3 microprocessor, four-channel relay, switches, sockets, power adapter and current sensors. Arduino is connected to the simulation environment via a universal serial bus and receives serial input. A preloaded program processes the information and the microprocessor issues a signal to a four-channel relay depending on the current environment. Current sensors are used to read current from appliances and determine their operation status. A power adapter is used as an external power supply source for the relay. The testing circuit developed in Fritzing software is given in Appendix: II.

3.9 Conclusion

This chapter discussed the development and execution of the research framework. The problem formulation was the minimization of electricity cost for a residential system. The problem was viewed as a Markov Decision Process consisting of the appliances as the environment, load management strategies as a set of actions, a single agent to manage the environment and fuzzy rule-based numerical rewards. An architecture showing the integration of the algorithm in a smart home energy management system was developed. For discrete states, fuzzy logic rules were adopted in the estimation of numerical rewards based on load demand and electricity tariff. The time of use (ToU) tariff was designed by benchmarking with other countries such as Italy and Australia. The ToU structure had three major price segments (off, mid and peak). A knowledge base was introduced to log in human feedback and update fuzzy logic rules accordingly purposely to integrate human dissatisfaction in the algorithm. Q-learning algorithm was designed with an agent that explored and exploited a set of actions through an e-greedy policy. This approach avoided biases by the agent in terms of decision-making based purely on maximizing rewards. The algorithm was tested in simulation and physical environments.

Chapter 4 Results and Discussion

4.1 Introduction

Recall the objective of the research as the design and testing of an algorithm that aims to minimize energy costs by applying effective load management strategies depending on a static time-of-use tariff plan, subsequently deploying it for testing in a physical system and consumer feedback integration. The key assumption is demand response (DR) is fully automated and that the agent makes the decisions with few human interruptions on the basis that human preference is already factored during load control categorization.

The results presented intend to answers the research questions such as the impact of the algorithm in costs savings or expense at a household level in respect to DR, the agent's learning curve against the training episodes to determine whether the gap on the curse of dimensionality is addressed, consumer feedback integration and a balance in load management to ensure the algorithm maintains similar an overall load curve area before and after demand response. The theory of learning curve is used to evaluate the performance of the algorithm during training by comparing it with empirical forms. The goal of the learning curve is to define points of convergence and verify that the algorithm has had sufficient training about the environment.

A residential load curve was collected from an operational mini-grid and scaled as input data to the algorithm. Testing of the algorithm in physical systems is also evaluated on an offline mode. An online system would require an advanced smart national grid which currently is not the case with Kenya's national grid. The overall effect of load shifting is additionally reviewed at a county level while considering simple assumptions on population and household electrification rate.

Data on appliance rating and time-of-use was collected from one household and categorized as illustrated in Table 4-1. This data was used to generate the load curve for input into the algorithm. Two households (data from operational mini-grid and constructed load profiles) are concurrently studied. The objective of the comparison is to analyze the agent's decision-making process and whether the results are predictive or assume a similar pattern in terms of the optimal policy from a set of load management strategies.

The Residential consumers' feedback on the Time of Use tariff plan is also analyzed as a preliminary social scoping to obtain the range of savings that residential consumers would prefer to voluntarily participate in DR programs.

The definition of appliance categories in the table below is defined in section 3.2.1.

Table 4-1 Appliance rating and cumulative loads

No.	Appliance	Category	Qty	Power Rating (W)	Total Load (W)	Cumulative load (W)
1.	Lighting	Non-schedulable	6	18	108.00	108.00
2.	Security Light	Non-schedulable	4	50	200.00	308.00
3.	Phone Charging	Non-schedulable	2	15	30.00	338.00
4.	Fridge/freezer	Non-schedulable	1	58	58.00	396.00
5.	Oven	Non-schedulable	1	1050	1050.00	1446.00
6.	Television	Non-schedulable	1	68	68.00	1514.00
7.	Printing machine	Non-schedulable	1	15	15.00	1529.00
8.	Laptop	Non-schedulable	2	45	90.00	1619.00
9.	Blending machine	Non-schedulable	1	400	400.00	2019.00
10.	Washing machine	Non-interruptible	1	380	380.00	2399.00
11.	dryer	Non-interruptible	1	160	160.00	2559.00
12.	Vacuum cleaner	Interruptible	1	700	700.00	3259.00
13.	Water heater	Interruptible	1	2200	2200.00	5459.00
14.	Iron	Interruptible	1	1700	1700.00	7159.00
15.	Juicer	Interruptible	1	500	500.00	7659.00
16.	Mixer	Interruptible	1	400	400.00	8059.00

4.2 Results and Discussion

4.2.1 Q-Learning Curve

Figure 4-1 expresses the learning curve as the graph of mean cumulative rewards as a consequence of the agent's optimal policy selection against the number of episodes or training time taken. The agent was trained using load demand and tariff data for five thousand iterations. After every 20 episodes, the Q-value table is greedily evaluated to obtain the mean cumulative reward for that episode.

One of the key challenges encountered in previous research work is the speed at which an agent learns the environment from a set of training data. It was observed that the agent converged after 500 episodes while considering consumer feedback integration on the optimal policy. This implies by setting the state-action pair space with a definite small size, the agent has the capability of learning from environment data and making an optimal decision within a smaller timeframe to attain convergence in a shorter episodic time. Another feature of the curve is a sharp convergence which illustrates low biases in learning and small variance in the mean cumulative reward at the convergence zone.

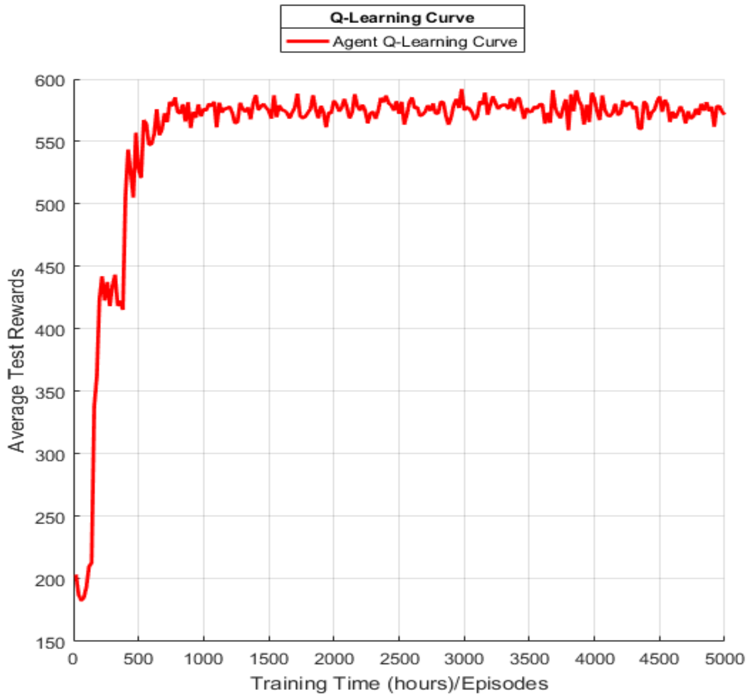


Figure 4-1 Training curve for the Algorithm

Consumers obtain optimal policy and can confirm their dissatisfaction level through feedback which is analyzed for cost implication. The agent of the algorithm has an improved learning speed even when consumer feedback integration is considered.

4.2.2 Demand Response

4.2.2.1Input Load Curves

The initial load curves are given in Figure 4-2. These are loads that are to serve as input into the algorithm. Residential consumer 1 and 2 load profiles represent data from operational mini-grid and constructed from time of use for appliances in Table 4-1, respectively. A possible approach to evaluate the overall effect of the agent's action is through an overlay of the recommended actions on the initial load curves described above. From Figure 3-2, valley-filling involved load addition while load shifting and clipping reduce certain loads from the load curve.

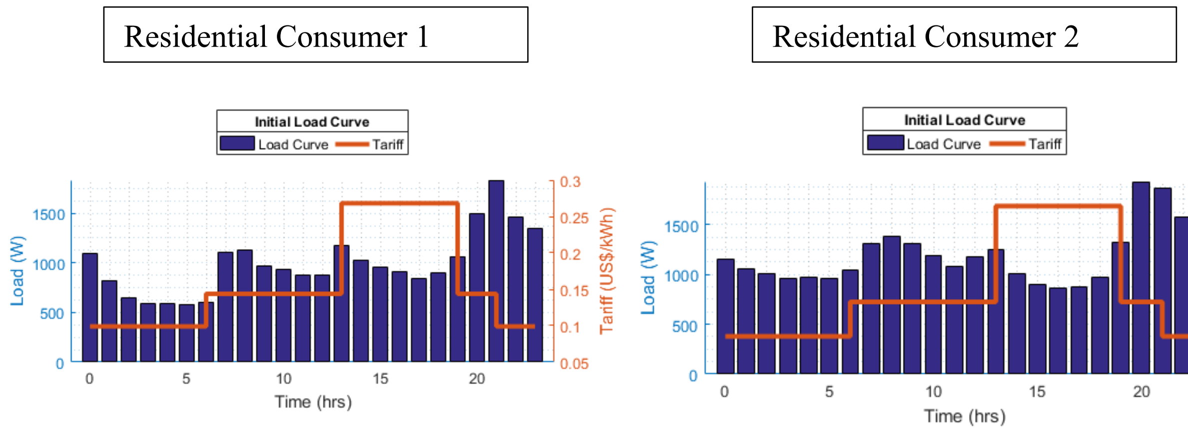


Figure 4-2 Input load curves for analysis by the algorithm

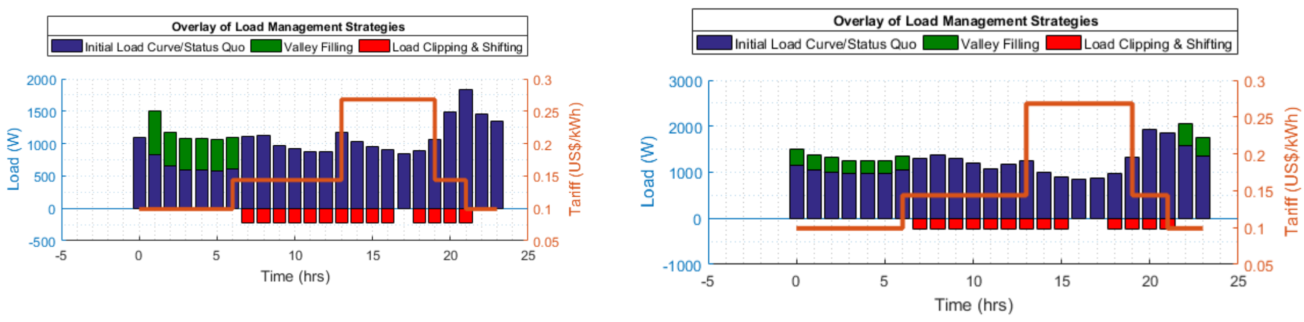


Figure 4-3 Overlay of load management strategies

4.2.2.2Modified Load Curves

Modified load curves are as a result of the load management strategies. Figure 4-4 illustrates the net load curve from the effect of the load management strategies on the initial load curve.

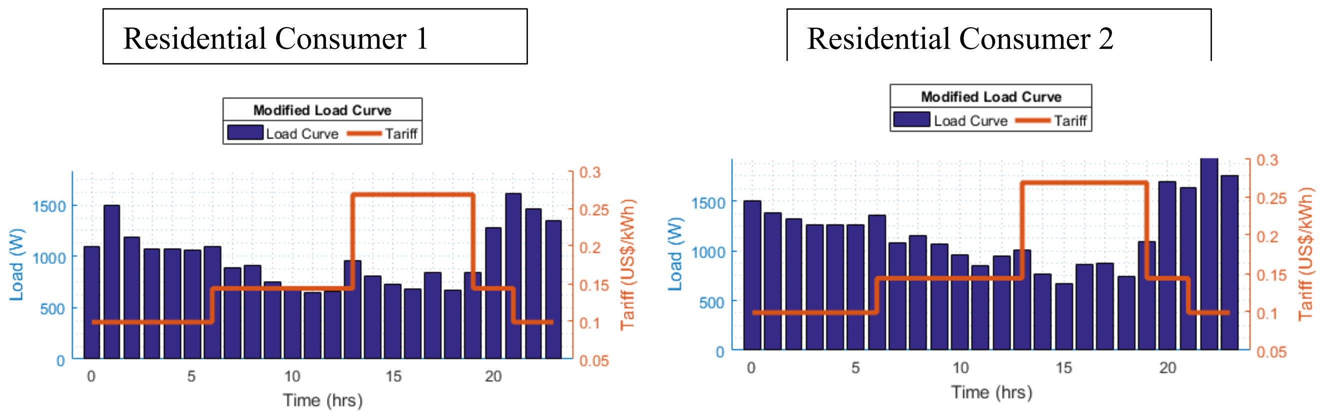


Figure 4-4 Demand response modified load curves

4.2.2.3Load Management Strategies

Individual load management strategies are presented to assess the time when they are applied. The sum of areas for respective load management strategies needs to be zero to confirm that all load shifted or clipped are compensated during off-peak hours.

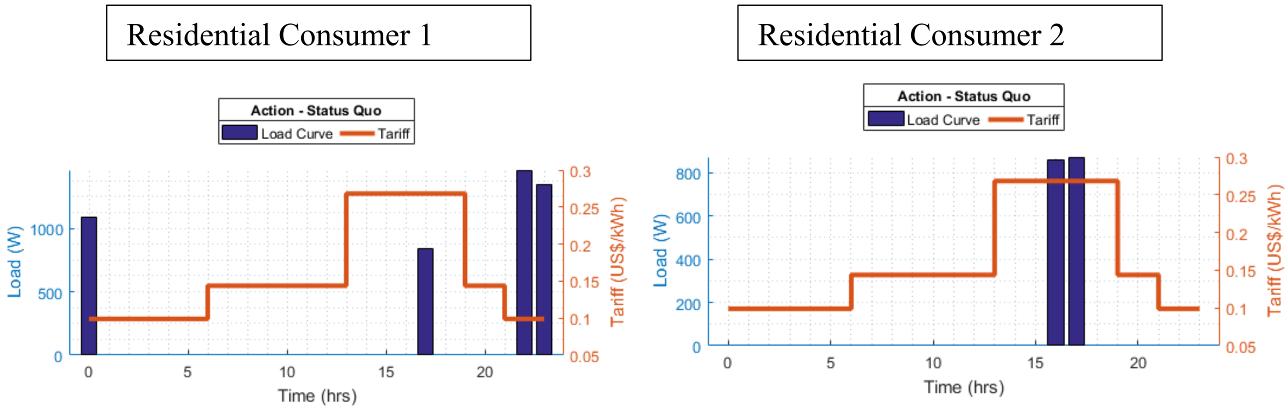


Figure 4-5 Load curve modification due to status quo action

The algorithm applies the status quo at 17:00 and between 22:00-00:00 HRS for residential consumer 1 while similar action between 16:00 and 17:59 HRS for residential consumer 2 as illustrated in Figure 4-5.

Figure 4-6, it was realized that both consumers never experienced peaked usage and therefore load clipping action never activated. Ideally, it would be rare for consumers to have over ninety percent of their total load active.

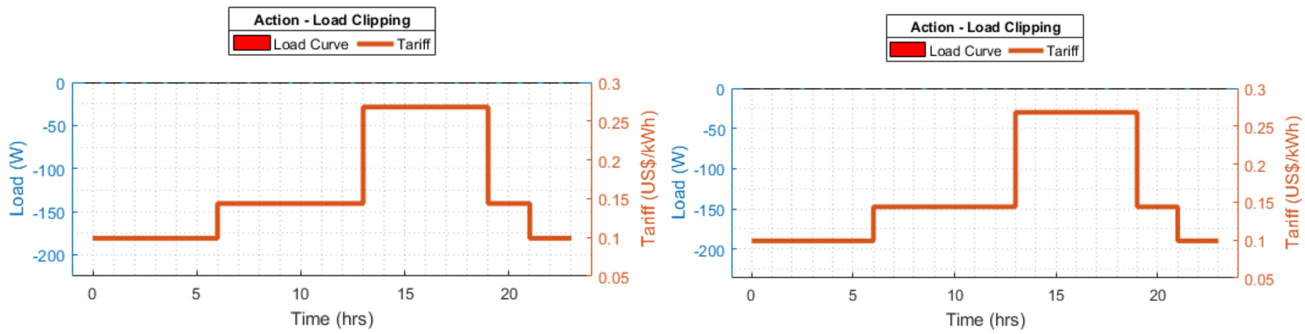


Figure 4-6 Load curve modification due to load clipping action

Load shifting is the active load management strategy, unlike load clipping. As shown in Figure 4-7, the agent feels an optimal action is to shift certain loads when electricity prices are high.

Recall the criteria behind action selection is a combination of load demand and electricity price at that time.

Valley-filling performs compensation of loads that were shifted or clipped as illustrated in Figure 4-8. The agent considers it optimal to perform this action when electricity prices are at the lowest.

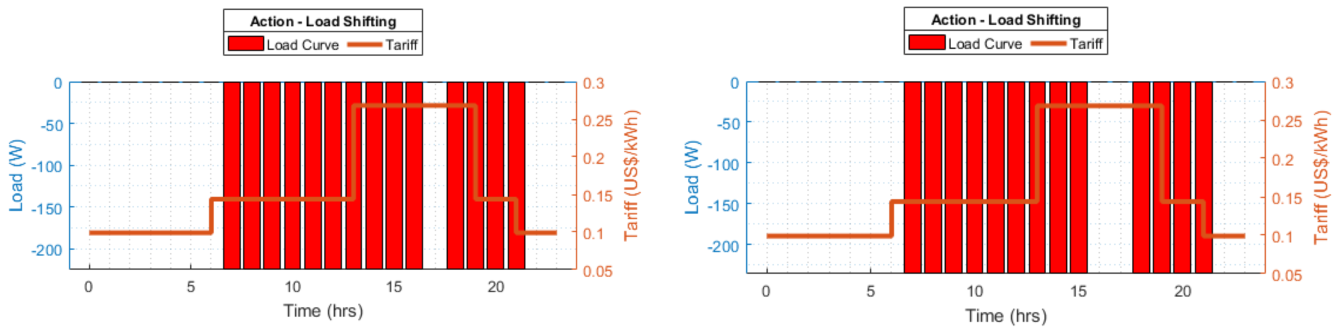


Figure 4-7 Load curve modification due to load shifting action

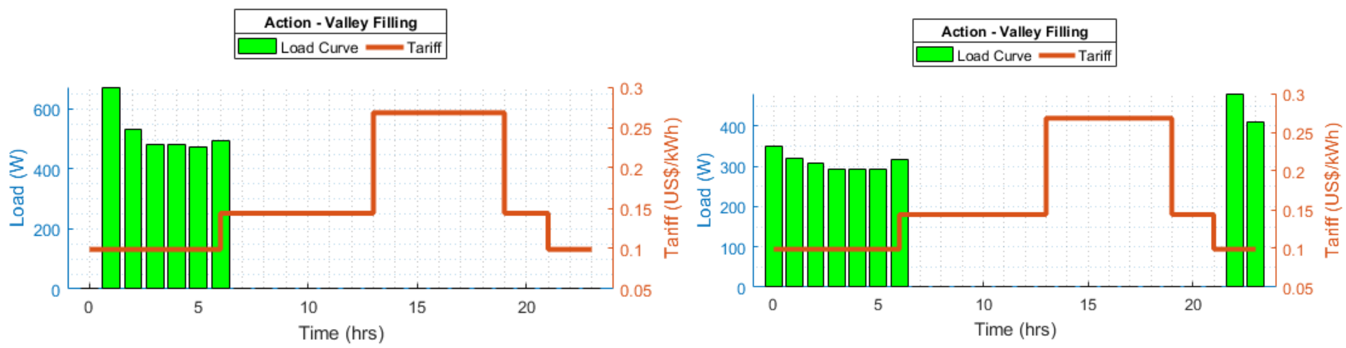


Figure 4-8 Load curve modification due to valley-filling action

4.2.3 Energy Economics

4.2.3.1 Energy Cost

Demand Response (DR) implemented by the algorithm shows an increase in energy cost for residential consumers 1 and 2 during valley filling while cost reduction during load clipping and shifting as illustrated in Figure 4-9 and Figure 4-10. For residential consumer 1, the applicable load management strategies include load shifting, valley-filling and status quo. Appliances in load category 3 were shifted during the peak times when the tariff is high. This happens between 07:00-16:00 and 18:00-21:00 HRS which are the mid-peak and peak hours.

Additional exploration is to compare the costs and savings as a result of the corresponding applicable load management strategies only at the region or time when they are applied. For the status quo, the net cost is zero since no action is taken. Figure 4-10 shows that during valley filling, the demand response effect is an additional electricity cost on the consumer since appliances are added hence an increase in total load. However, the major impact is in terms of net savings. The algorithm observes this region as either point of low tariff high load or high tariff low load.

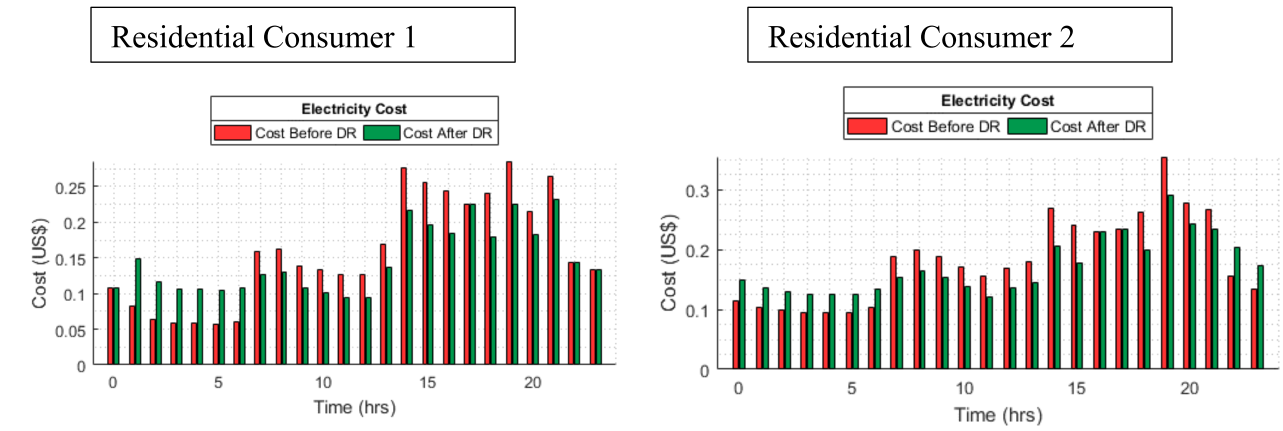


Figure 4-9 Comparison of energy costs before and after demand response

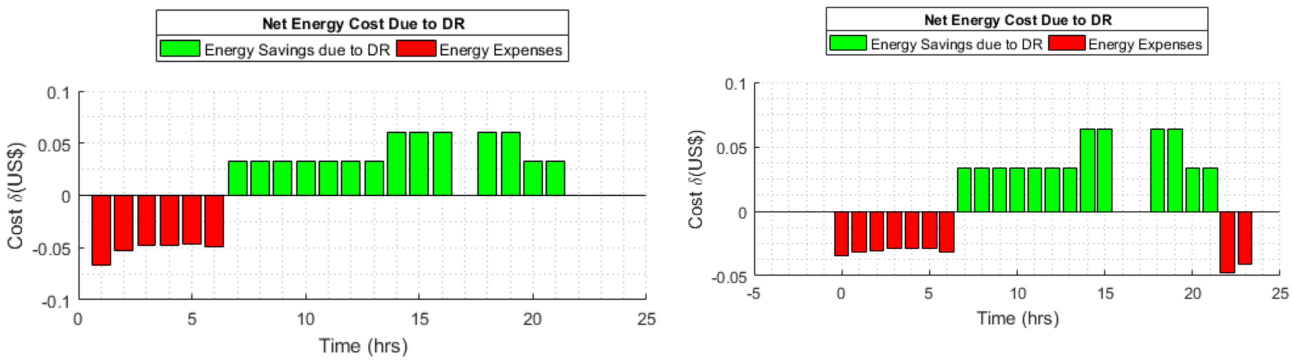


Figure 4-10 Regional effect of demand response on energy cost and savings

As illustrated by the left images of Figure 4-11, the action of load shifting and clipping strategies saved 19% of the initial energy expense while valley filling resulted in an additional cost of 8%. The net savings for residential consumer 1 as a result of the agent's optimal policy response is 11%. Residential consumer 2 realized cost savings of 14% during load shifting and clipping strategies while an additional cost of 6% during the valley filling regime. The net savings for residential consumer 2 as a result of an optimal policy by the agent is 8%.

4.2.4 High-Level Effect of Demand Response at County Level

The value of demand response may be perceived as insignificant at the individual residential level, however, cumulatively at grid level, it bears significant load savings. Utilities have the potential of serving other loads using the same existing infrastructure. This study demonstrates at a high-level, the effect of individual participation in demand response subject to the following assumptions; the socio-economic characteristics of the County include an electrification rate of 83.971 % [43], 1,506,888 households [44] out of which 40% participate in the ToU plan.

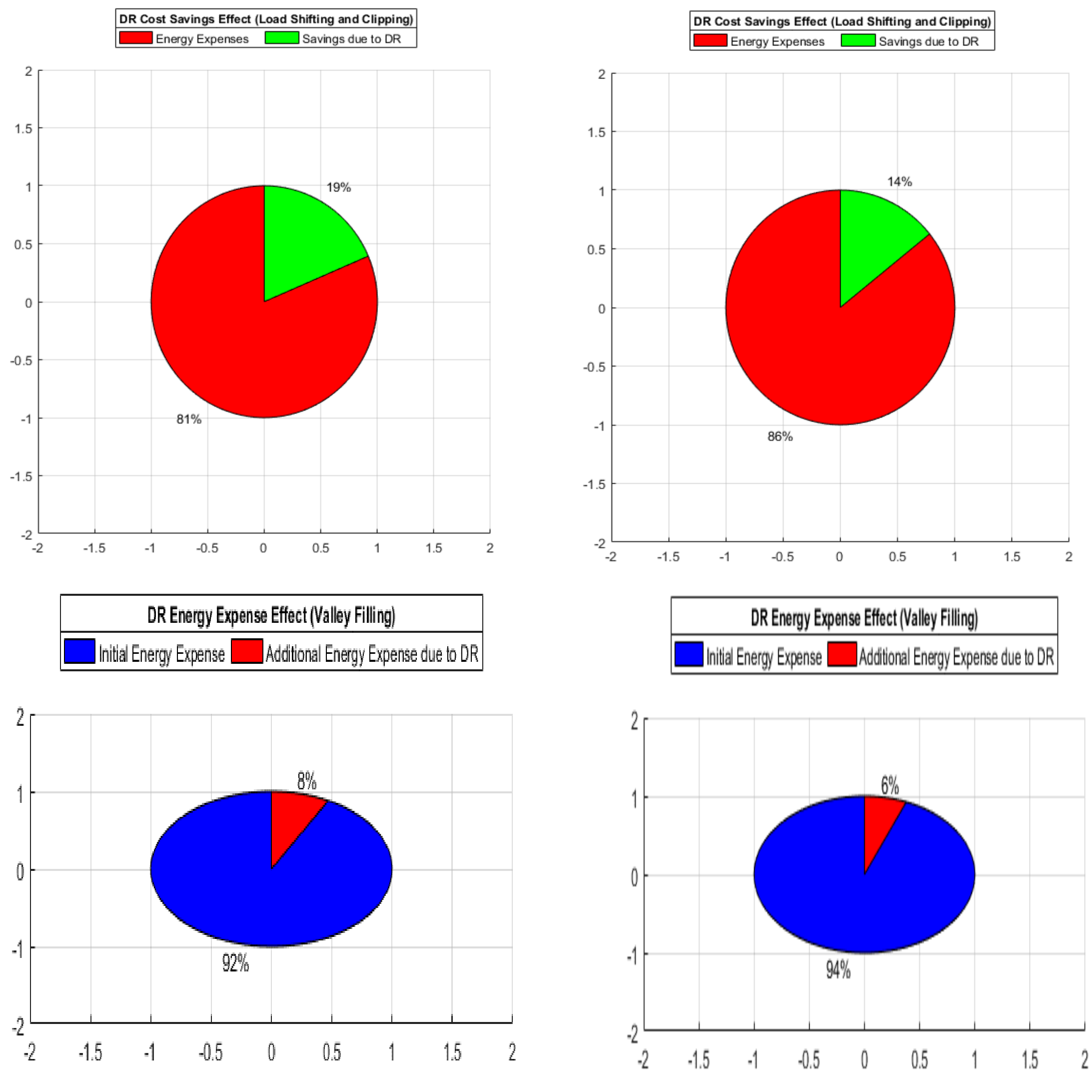


Figure 4-11 Energy expense and savings as a percentage of initial energy cost

The findings are load clipping and shifting of residential loads by the algorithm could provide 110 and 130 MW capacity during peak-times if the average load curves for all residential systems corresponded to those of residential consumer 1 and 2, respectively as illustrated in Figure 4-12. Recall that this capacity is deployed to commercial and industrial consumers who are actively consuming energy from the grid at this time. At this period residential consumers earn incentives from utilities depending on their participation level. The tradeoff is a possible deferment of additional capacity by residential consumers while they receive incentives in terms of discounted tariffs at low-peak periods. As such, the electricity tariff becomes affordable due to the management of demand and supply. Also, there is a great possibility of utility peak reduction.

Average residential consumption 1

Average residential consumption 2

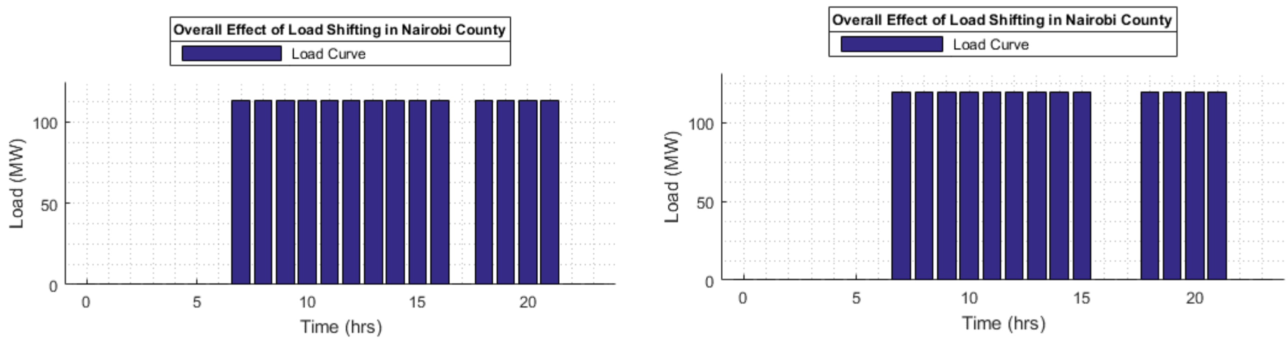


Figure 4-12 Effect of demand response at grid scale

4.2.5 Time of Use Survey for Residential Consumers

Demand response can be voluntary or mandatory which is dependent on the conditions of a country or utility. This research considered a voluntary approach towards demand response, therefore, it was invaluable to obtain preliminary feedback from few residential consumers with regards to demand response through a short survey. The majority of residential energy consumers preferred a Time of Use (ToU) plan if they were to realize significant savings above 20 percent. The willingness to install smart home energy management systems in their systems is dependent on guarantee savings on energy bills.

In the context of technology, part of residential consumers felt that switching to ToU structure will require sophisticated metering and billing systems and the cost of transition would be loaded to consumers making electricity tariffs go beyond the current unbearable prices.

4.2.6 Q-Learning Algorithm Testing

A user-defined graphical user interface (GUI) designed in Matlab app designer software was used to provide a consumer interaction with the optimal policy of the algorithm during testing on the physical system. The user was advised on the current tariff and how the loads are scheduled for 24 hours. The performance of the algorithm through the GUI is as shown in Figure 4-14. When the tariff is low, the algorithm sends a serial command to Arduino hardware to close a normally open (NO) relay for all four channels

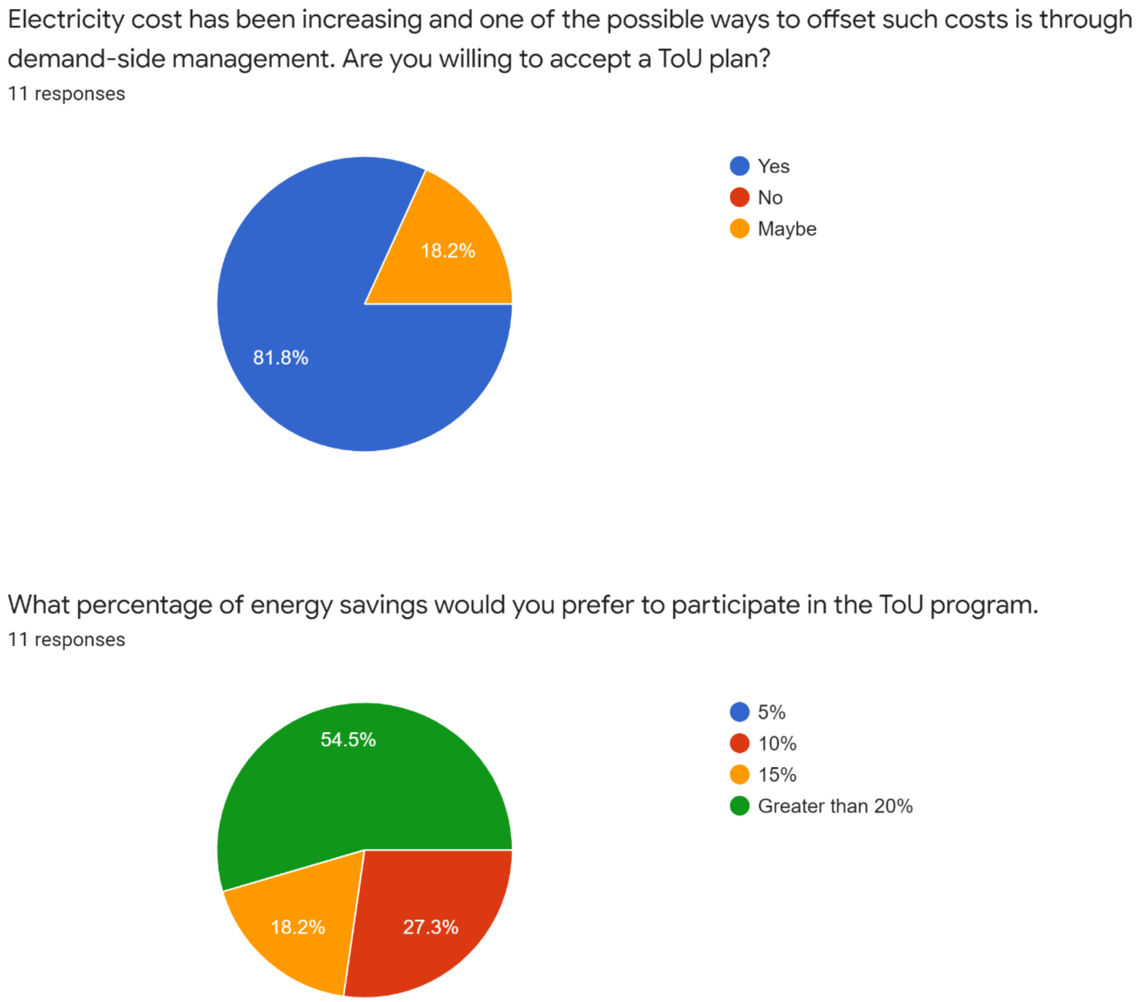


Figure 4-13 Results of Time of Use survey

However, the consumer is given the option to either valley-fill appliances in category 3 or 4. For example, the first figure confirms the algorithm action during valley filling when the consumer adjusts the discrete knob to “IP4”, the command is close the NO relay for channel four to turn on appliances in category four while turning off appliances in category three purposely to avoid peaking.

The algorithm considers load shifting of appliances in category 3 when tariffs are higher and depending on the time of use maintain a status quo of the appliances. At status quo, the algorithm inspects the status of the environment and issues no command but rather through its control objective ensures the status quo action is truly achieved.

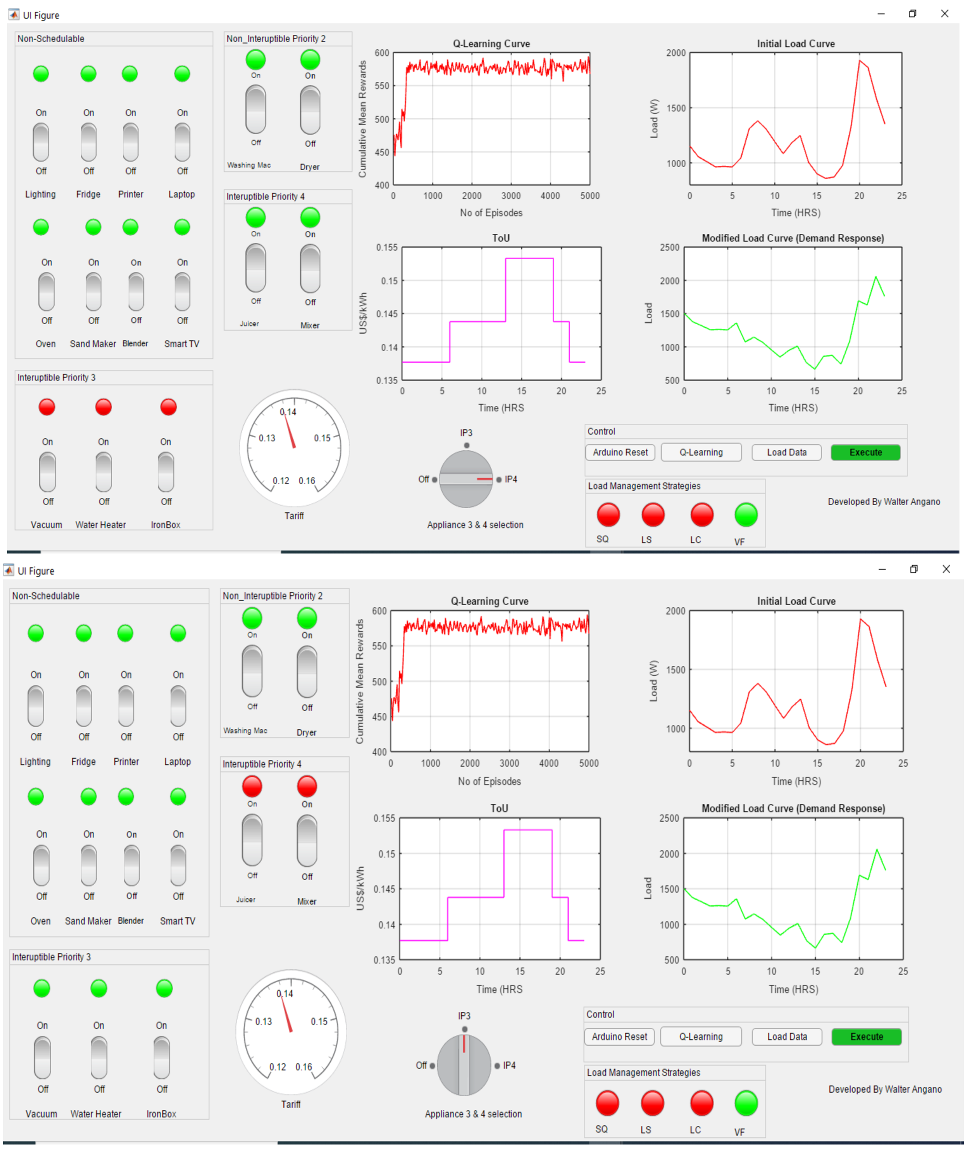


Figure 4-14 GUI interfaced with the testing system

4.3 Validation of Results

The research results are validated two-fold which are a comparison of the learning time with other publications and computation of relative error curve. Comparison of percentage energy savings results was not considered in the validation as figures vary by country depending on the time of use tariff and other financial incentives.

Research by [30] applied deep reinforcement learning in optimal scheduling of residential appliances with a configuration consisting of neural network-based optimal policy. Convergence was achieved after 1500 iterations. Authors in [27] who adopted fuzzy reasoning in reinforcement learning achieved convergence after 10,000 iterations. Other algorithm designs converged after 200, however for collective load systems exclusive of individual appliances. The convergence of the Q-learning algorithm for this research begins at episode 500 and normalizes at episode 750.

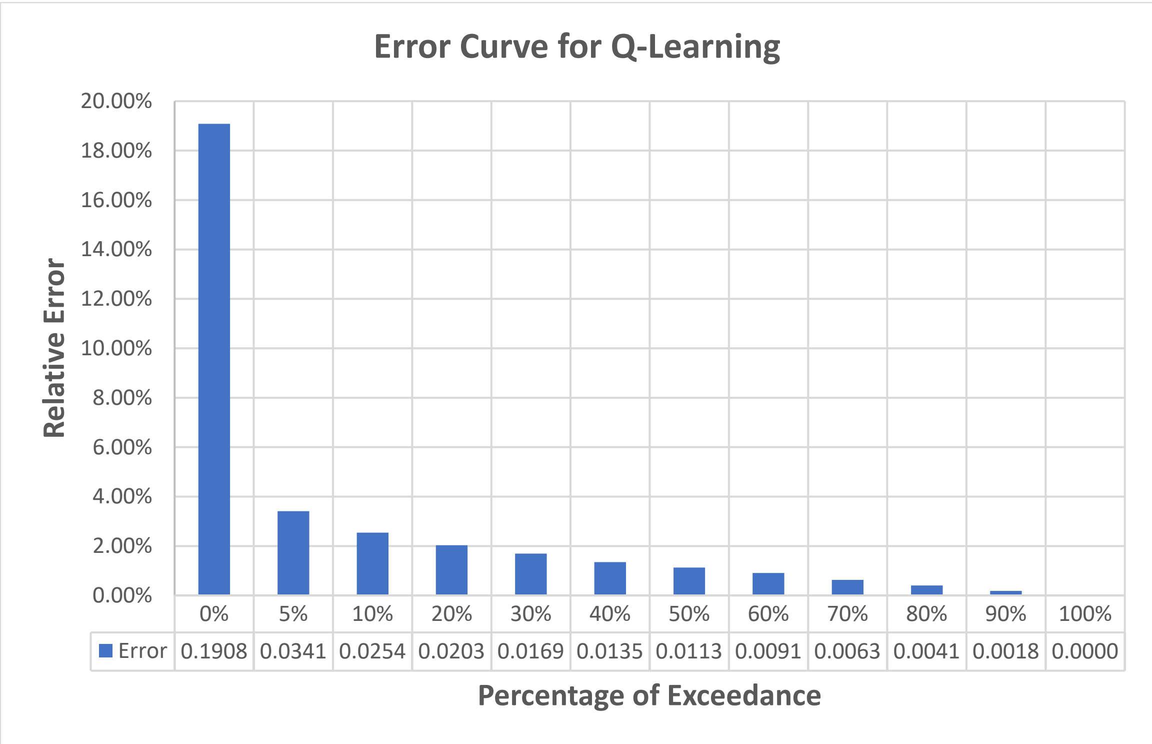


Figure 4-15 Error curve for Q-learning algorithm

Learning episodes bear a variation depending on the methodology that authors develop. It may be concluded that the learning speed falls within a range with other previous findings, however, additional analysis is done to evaluate the relative error per episode.

Figure 4-15 illustrates the error curve during the learning of the agent. The learning error reduces as the agent learns the environment, for example, 80 % of the time, the algorithm will bear a relative error of 0.0041 when making determining the optimal policy. The error declines with an increase in the learning period.

4.4 Conclusion

This chapter presented the findings of the research and discussion of the algorithm learning curve. The learning speed of the urgent is improved if the state-action space is reasonable enough. In this case, the state-action space was eight which was dependent on load categorization by the consumer. The Algorithm commenced convergence within 500 episodes and stabilized upon reaching 750 episodes. These values were found to be within range when compared with previous research work. Results were validated by comparison of training episodes at convergence with related research works. Additionally, the relative error was computed per episode to investigate the exceedance levels. The purpose of such validation is to determine relative error at particular percentages of exceedances. Improvement is envisioned in learning particularly when using continuous improvement fuzzy logic rules. While demand response seems to have insignificant energy savings, its effect at the grid level is significant. Residential consumers prefer significant energy savings, greater than 20 percent of their energy bills, to voluntarily participate in demand response programs. The effect of demand response at the residential level is insignificant. However, the effect at the grid level is significant with the potential of deferring capital-intensive energy generation systems. In terms of acceptance of such programs, residential consumers prefer higher energy savings or strong financial incentives to participate.

Chapter 5 Conclusion and Recommendations

5.1 Research Conclusion

There has been an evolution in algorithms that implement demand response in smart home energy management systems. Q learning is a predominant tool of reinforcement learning that researchers have found resourceful when establishing optimal policy from a set of actions. However, one of the major research gaps in previous studies was the curse of dimensionality where the agent's learning speed is lower and takes time to determine an optimal policy. Also, other authors used traditional means of classifying load limits with limited consumer preference in selecting priorities.

This research concurrently addressed the gaps by establishing a state-space action consisting of four load management strategies which are status quo, valley filling, load shifting and load clipping. The study also allowed the consumer to group appliances according to their priority and usage frequency, subsequently sets load control limits that act as state space. A fuzzy logic system estimated the rewards associated with each action as output by taking load demand and electricity tariff as its input. A knowledge improvement base was developed to update the fuzzy rules and ensure the algorithm minimizes consumer dissatisfaction. This approach resulted in a reduced learning speed by the agent with convergence commencing in 500 episodes and stabilizing in 750 episodes.

A testing system was assembled and interfaced with a graphical user interface designer using app designer in Matlab. Through serial communication, the Arduino microprocessor received command signals from Matlab and based on a pre-defined interpretation either activated or deactivated a relay to turn the loads on or off. To achieve this communication, a program was preloaded in Arduino with each time confirming the status of the serial port from Matlab. It is considered possible to set-up and a smart home energy management system using cheaper kits such as Arduino.

The designed algorithm has the potential of minimizing energy costs depending on the load curves. For example, 8 and 11 % of energy cost savings were realized for residential consumers 2 and 1, respectively.

Research results were validated by comparing the algorithm's convergence with those achieved in previous studies and generating relative error per episode.

5.2 Research Recommendations

According to [35], Kenya's energy supply is envisaged to exceed demand due to excess generation in the coming years. The updated Least Cost Power Development Plan (LCPDP) report indicates a revised commercial operation date for small hydropower projects with most being pushed up to ten years. The report provides recommendations for creating stimulation for the consumption of excess energy.

However, electricity tariffs since 2018 illustrate an increasing tariff trend in Kenya which may limit economic growth due to expensive power. Heavy consumers are now opting for investment in their generation to offset huge energy costs and improve their future cash flows. Such tremendous shifts to self-generation by anchor consumers translates to an increase in electricity tariffs which hinders access to energy not due to proximity with the distribution grid but as a result of the inability to pay for expensive power. The predictive analysis is such consumers may develop a local energy market pool where they can trade amongst themselves.

With a reduction in bulk consumption, residential consumers are slowly evolving into anchor customers. Probably with increasing tariff, a greater percentage might shift to self-generation using alternative sources of energy such as solar photovoltaic technology. One of the recommendations outlined in the LCPDP report is demand-side management, particularly load shifting, can promote optimal energy consumption by expanding the Time of Use (ToU) concept. However, considering the historical trend in tariff and modeling it as a ToU, this research determines that savings by residential consumers may vary. Some residential consumers may realize daily savings of 3 percent others 5 percent which depends on individual load demand. Residential consumers may also find it uncertain to change their consumption patterns and this may call for additional investment in energy storage systems which requires a significant incentive by utilities.

The implementation of ToU programs requires advanced technology. This includes advanced smart metering infrastructure, well-established communication protocols and software systems with algorithms that can respond automate and adjust energy consumption. To realize demand-side management in the context of demand response, this research recommends the following:

- Utilities should provide a strong incentive to demonstrate significant savings by residential consumers. Also, savings need to be certain otherwise uncertainty introduces frustration in demand response participation. Government subsidies are essential to ensure the cost of switching to the ToU plan is not loaded to residential electricity tariff.

- There are still limited studies in demand response particularly in the Kenyan context with regards to both technical and social scope. Social scoping for demand response should target to investigate public view and obtain feedback on incentives that consumers anticipate in such programs.
- A legal framework is essential in encouraging the incorporation of energy management services that run cloud computing infrastructure such as Microsoft Azure in deploying algorithms and managing individual residential energy management systems. This will stimulate growth and competition in the provision of energy management systems.
- There is a need for marketing demand response programs and encourage research through collaborative partnerships with higher learning institutions who through research aid in developing tailored applications.

This research considers the following are potential research improvements

- Integrating consumer feedback as a crisp input and developing dissatisfaction models. Consumer dissatisfaction models are developed from historical feedback data and incorporated in fuzzy rules.
- Cloud services technologies can be integrated to develop dissatisfaction models for per residential system and effectively manage optimal policies by integrating such models as fuzzy systems.

5.3 Research Contribution

This research integrated human feedback in its fuzzy control rule system through a graphical user interface. This was achieved using a knowledge base where feedback is compared to the optimal policy of the algorithm. The load-electricity tariff grid was developed by considering the cumulative load of the appliances arranged according to their category. When classifying loads, rather than assuming traditional methods and considering that load profiles vary with individual consumers, the cumulative load levels at each category were adopted as a load control level (LCL). These LCLs together with tariff data were considered as the key elements of the load-electricity grid. A fuzzy logic control system was applied when approximating rewards for a particular state and action. To further prove the automation of demand response (DR) using cost-effective tools, the algorithm was tested in a physical environment consisting of Arduino Uno microprocessors, current sensors, relays, switches and loads. The optimal policy was generated in Matlab software and instructions were issued through serial communication to manage the loads.

References

[1] United Nations Industrial Development Organization, "Sustainable Energy Regulation and Policy-making Training Manual - Demand Side Management," United Nations Industrial Development Organization, 2009.

[2] I. Hussain, S. Mohsin, A. Basit, Q. U. Khan ZA and N. Javaid, "A review on demand response: pricing, optimization, and appliance scheduling," *Procedia Computer Science,* vol. 52, p. 843–50, 2015.

[3] L. Summit Blue Consulting, "Evaluation of the 2006 Energy-Smart Pricing Plan," Chicago, 2007.

[4] K.P.L.C, "The Kenya Power and Lighting Company Limited - Annual Report and Financial Statements 2017/2018," Kenya Power & Lighting Limited, Nairobi, 2018.

[5] Energy and Petroleum Regulatory Authority, "Updated Least Cost Power Development Plan," Government Printer, Nairobi, 2018.

[6] Energy and Petroleum Authority, "Energy and Petroleum Authority," 06 06 2018. [Online]. Available: https://www.epra.go.ke/wp-content/uploads/2018/06/Electricity-tarrifs-for-the-2018-2019-tarrif-control-period.pdf. [Accessed 28 08 2020].

[7] Z. Zillman, J. González, L. Godde and I. Guayo, Energy Justice and Energy Law, Oxford: Oxford University Press, 2020.

[8] J. Vázquez-Canteli and Z. Nagy, "Reinforcement learning for demand response: A review of algorithms and modeling techniques," *Applied Energy,* vol. 235, pp. 1072-89, 2019.

[9] The Association of German Engineers, "VDI-Guideline VDI 4602," Berlin, 2007.

[10] C. B. Smith and K. E. Parmenter, "Energy Management Principles: Applications, Benefits & Savings," in *Chapter One Introduction*, Amsterdam, Elsevier, 2015, pp. 1-15.

[11] C. W. Gellings, "Demand-side management:," Vols. I-V, 1988.

[12] Y. D. Goswami and F. Kreith, "Demand-Side Management," in *Energy Efficiency and Renewable Energy Handbook*, New York, CRC Press, 2016, pp. 289-294.

[13] J. M. Melillo, T. Richmond, Richmond and G. W. Yoh, "Highlights of Climate Change Impacts in the United States: The Third National Climate Assessment. U.S. Global Change Research Program," U.S Government Printing Office, Washington, 2014.

[14] W. C. Turner and S. Doty, Energy Management Handbook, Lilburn: The Fairmont Press Inc, 2007.

[15] C. Goldman, R. Levy and A. Silverstein, "National Action Plan for Energy Efficiency : Coordination of Energy Efficiency and Demand Response," U.S. Department of Energy, 2010.

[16] The US Department of Energy, "Benefits of Demand Response in Electricity Markets and Recommendations for Achieving Them. A report to the United States Congress Pursuant to Section 1252 of the Energy Policy Act of 2005," Washington, DC: Government Printing Office., Washington, 2006.

[17] International Renewable Energy Agency, "Innovation landscape brief: Time-of-use tariffs," IRENA, Abu Dhabi, 2019.

[18] B. Mahapatra and N. A, "Home energy management system (HEMS): concept, architecture, infrastructure, challenges and energy management schemes," *Energy Systems,* 2019.

[19] B. Asare-Bediako, W. Kling and P. Ribeiro, "Home energy management systems: Evolution, trends and frameworks," *2012 47th International Universities Power Engineering Conference (UPEC),* pp. 1-5, 2012.

[20] B. Zhou, W. Li, W. K. Chan, Y. Cao, Y. Kuang, X. Liu and X. Wang, "Smart Home Energy Management Systems: Concept, configurations and scheduling strategies," *Renewable and Sustainable Energy Reviews,* pp. 30-40, 2016.

[21] A. Nacer, B. Marhic and L. Delahoche, "Smart Home, Smart HEMS, Smart heating: An overview of the latest products and trends," *2017 6th International Conference on Systems and Control (ICSC),* pp. 90-95, 2017.

[22] EDUCBA, "Machine Learning Algorithms," [Online]. Available: https://www.educba.com/machine-learning-algorithms/. [Accessed 17 December 2020].

[23] L. Andrea, "The Landscape of Reinforcement Learning," in *Reinforcement Learning with Python*, Birmingham, Packt Publishing Ltd, 2019, pp. 9-23.

[24] R. S. Sutton and A. G. Barto, "Reinforcement Learning," in *Finite Markov Decision Processes*, Cambridge, The MIT Press, 2018, pp. 47-71.

[25] S. Zhou, Z. Hu, W. Gu, M. Jiang and X. Zhang, "Artificial intelligence based smart energy community management: A reinforcement learning approach," *CSEE Journal of Power and Energy Systems,* vol. 5, pp. 1-10, 2019.

[26] S. Lee and D. Choi, "Reinforcement Learning-Based Energy Management of Smart Home with Rooftop Solar Photovoltaic System, Energy Storage System, and Home Appliances," *Sensors,* vol. 19, p. 3937, 2019.

[27] 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.

[28] M. Khan, J. Seo and D. Kim, "Real-Time Scheduling of Operational Time for Smart Home Appliances Based on Reinforcement Learning," *IEEE Access,* vol. 8, pp. 116520-116534, 2020.

[29] X. Xu, Y. Jia, Y. Xu, Z. Xu, S. Chai and S. C. Lai, "IEEE Transactions on Smart Grid," *A Multi-Agent Reinforcement Learning-Based Data-Driven Method for Home Energy Management,* vol. 11, pp. 3201-3211, 2020.

[30] H. Li, Z. Wan and H. He, "Real-Time Residential Demand Response," *IEEE Transactions on Smart Grid,* vol. 11, pp. 4144-4154, 2020.

[31] Regulus Limited, "Electricity cost in Kenya," December 2020. [Online]. Available: https://stimatracker.com/historic. [Accessed 26 December 2020].

[32] L. X. Wang, A Course in Fuzzy Systems and Control, Prentice Hall PTR, 1997.

[33] B. Jang, M. Kim and G. Harerimana, "Q-Learning Algorithms: A Comprehensive Classification and Applications," *IEEE Access,* vol. 7, pp. 133653-133667, 2019.

[34] A. Lonza, "Reinforcement Learning Algorithms with Python," in *Q-Learning and SARSA*, Birmingham, Packt Publishing, 2019, pp. 74-97.

[35] Energy and Petroleum Regulatory Authority, "Updated Least Cost Development Plan Study Period 2020-2040," EPRA, Nairobi, 2020.

[36] J. Guerrero, A. Chapman and G. Verbic, "A Study of Energy Trading in a Low-Voltage Network: Centralised and Distributed Approaches," pp. 1-6, 2017.

[37] Energy Save, "Understanding PG&E's time-of-use rate options," [Online]. Available: https://news.energysage.com/pge-rate-schedule-increase-solar/. [Accessed 20 February 2021].

[38] Waterloo North Hydro Inc., "Residential Charges and Rates," 23 February 2021. [Online]. Available: https://www.wnhydro.com/en/your-home/Residential_Charges_and_Rates.asp. [Accessed 27 March 2021].

[39] AGL Sales Pty Limited, "AGL Freedom - Victoria residential electricity market offer - AGOMR00450," AGL Sales Pty Limited, Shorncliffe, 2018.

[40] Canstar Blue, "Peak and off-peak electricity times explained," 5 July 2020. [Online]. Available: https://www.canstarblue.com.au/electricity/peak-off-peak-electricity-times/. [Accessed 27 March 2021].

[41] Electricity Supply Board, "ESB Customer Supply Proposal for Smart Metering Tariff," ESB, South City, Ireland, 2010.

[42] Ceylon Electricity Board, "Residential and Religious," Ceylon Electricity Board, 12 September 2014. [Online]. Available: https://ceb.lk/tariff_catergory/en. [Accessed 27 March 2021].

[43] World Bank, "Access to Electricity, Urban Population - Kenya," 2018. [Online]. Available: https://data.worldbank.org/indicator/EG.ELC.ACCS.UR.ZS?locations=KE. [Accessed 2 Fenruary 2021].

[44] Kenya National Bureau of Statistics, "2019 Kenya Population and Housing Census," KNBS, Nairobi, 2019.

[45] United Nations Industrial Development Organization, "Sustainable Energy Regulation and Policy-making Training Manual - Demand Side Management," United Nations Industrial Development Organization, 2009.

[46] Electricity Supply Board, "ESB Customer Supply Proposal for Smart Metering Tariff," ESB, South City, Ireland, 2010.

Appendices

Appendix I: Originality Report

Design and Testing of a Demand Response Q-Learning Algorithm for a Smart Home Energy Management System

By: Walter Angano (F56/33270/2019)

Master of Science in Energy Management

Project Originality Report

Signed:

Student

Walter Angano	……………………………… Signature	19/08/2021 ……… Date

Supervisors

Prof. Cyrus Wekesa	……………………………… Signature	22/08/2021 ……… Date
Dr. Peter Musau Moses	……………………………… Signature	22/08/2021 ……… Date

Dean Faculty of Engineering

Prof. Ayub Gitau	……………………………… Signature	24.08.21 ……… Date

MSC Energy Walter Angano

ORIGINALITY REPORT

9%	6%	6%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	erepository.uonbi.ac.ke Internet Source	1%
2	cms.doe.gov Internet Source	<1%
3	Craig B. Smith, Kelly E. Parmenter. "Introduction", Elsevier BV, 2016 Publication	<1%
4	www.mathworks.com Internet Source	<1%
5	José R. Vázquez-Canteli, Zoltán Nagy. "Reinforcement learning for demand response: A review of algorithms and modeling techniques", Applied Energy, 2019 Publication	<1%
6	www.slideshare.net Internet Source	<1%
7	studentsrepo.um.edu.my Internet Source	<1%
8	Submitted to CSU Northridge Student Paper	<1%

9	Murad Khan, Junho Seo, Dongkyun Kim. "Real-Time Scheduling of Operational Time for Smart Home Appliances based on Reinforcement Learning", IEEE Access, 2020 Publication	<1 %
10	W.K. Wong. "A Fuzzy Capacity-Allocation Model for Computerised Fabric-Cutting Systems", The International Journal of Advanced Manufacturing Technology, 2003 Publication	<1 %
11	Submitted to University of Sheffield Student Paper	<1 %
12	Submitted to 7996 Student Paper	<1 %
13	upcommons.upc.edu Internet Source	<1 %
14	webapps.itc.utwente.nl Internet Source	<1 %
15	Hepeng Li, Zhiqiang Wan, Haibo He. "Real-Time Residential Demand Response", IEEE Transactions on Smart Grid, 2020 Publication	<1 %
16	nanapi.co.jp Internet Source	<1 %
17	hdl.handle.net Internet Source	<1 %

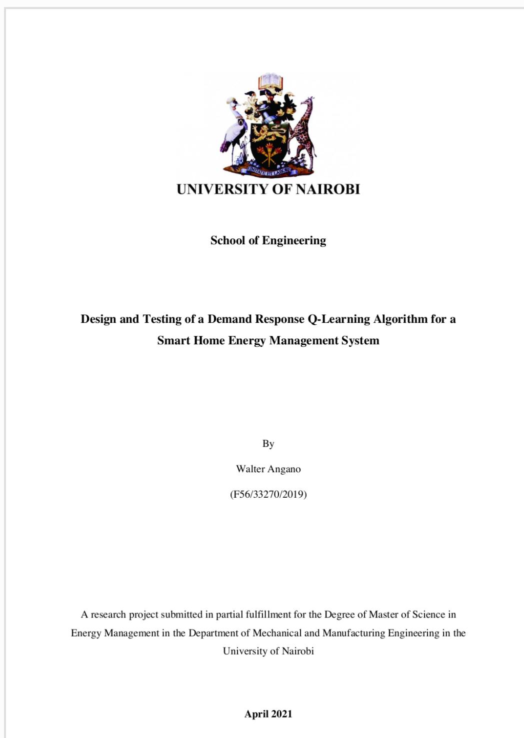
turnitin

Digital Receipt

This receipt acknowledges that Turnitin received your paper. Below you will find the receipt information regarding your submission.

The first page of your submissions is displayed below.

Submission author:	ANGANO WALTER LUSAGA
Assignment title:	MSC (ENERGY)
Submission title:	MSC Energy Walter Angano
File name:	F56_33270_2019_May_2021_Signed.pdf
File size:	3.5M
Page count:	108
Word count:	23,057
Character count:	135,875
Submission date:	23-Jun-2021 08:29PM (UTC+0300)
Submission ID:	1611182534



UNIVERSITY OF NAIROBI

School of Engineering

Design and Testing of a Demand Response Q-Learning Algorithm for a Smart Home Energy Management System

By

Walter Angano

(F56/33270/2019)

A research project submitted in partial fulfillment for the Degree of Master of Science in Energy Management in the Department of Mechanical and Manufacturing Engineering in the University of Nairobi

April 2021

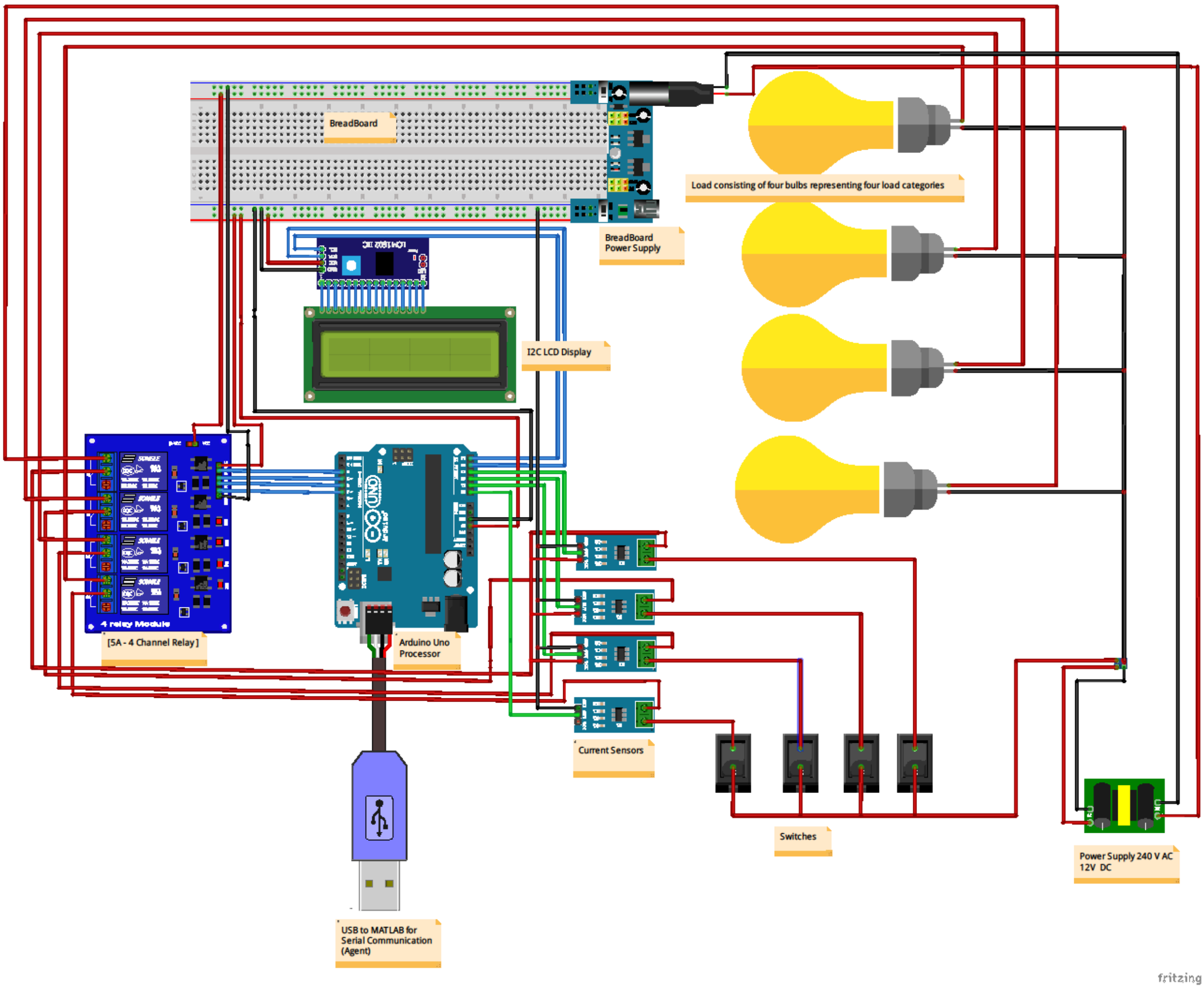
Appendix II: Amendment of Final Report

Per the defense presentation on the 18th of May 2021, the issue raised was the inclusion of the research gap table which has been addressed as illustrated in the changelog below.

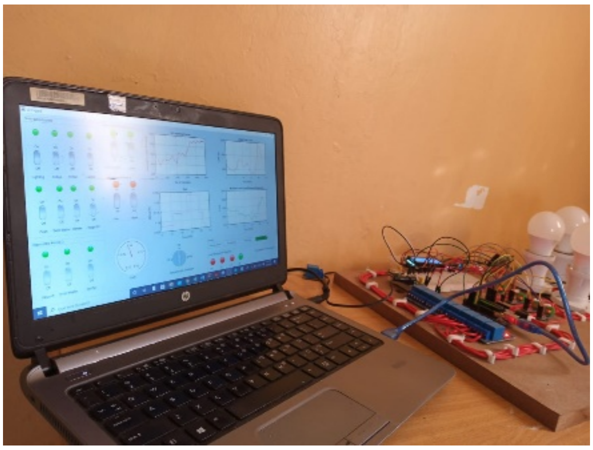
Table 0-1 Post-defense report changelog

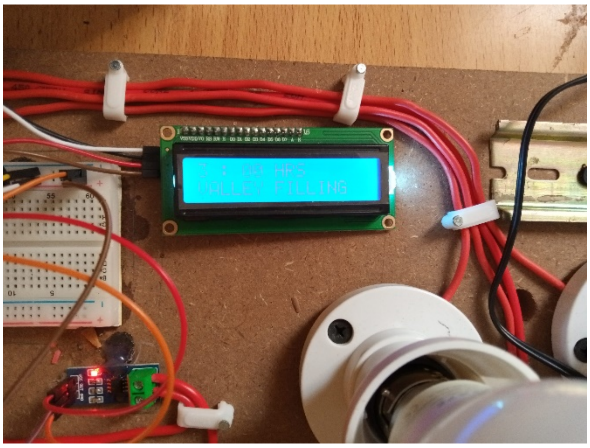
Date issue was raised	**Defense presentation Issue Raised**	**Report Log**
18th May 2021	Include research gap table	The research gap table is included as **Table 2-2** summarizes works from previous related research work.

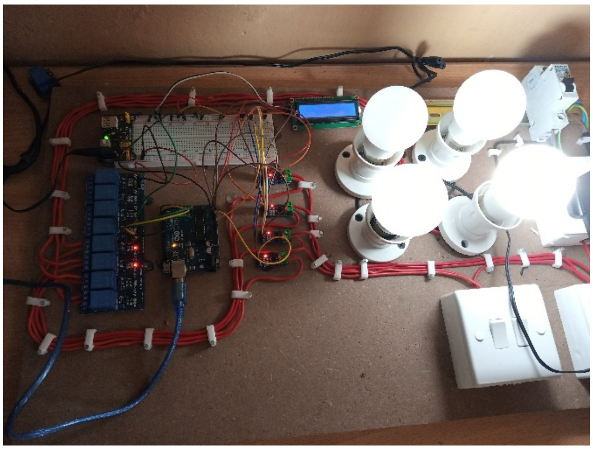
Appendix III: Testing Circuit Diagram

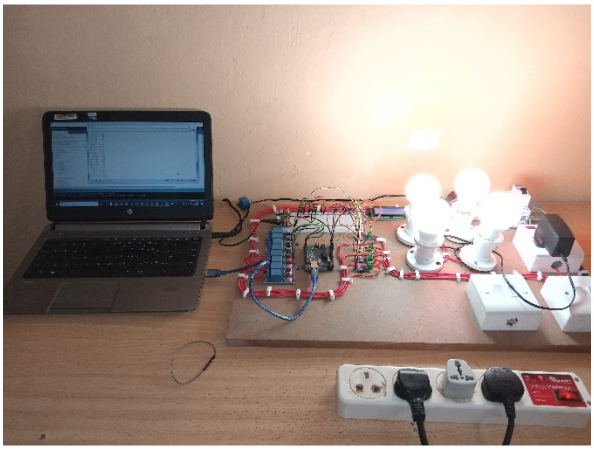


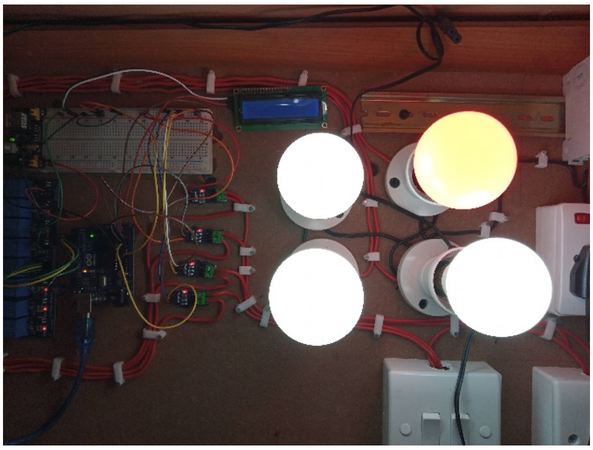
Appendix IV: Testing Hardware











Appendix V: Computer Programs

Matlab Computer Program

Main Function

```
% This is a Head program that calls the Q learning function.
% This code initializes the main parameters that are passed to the Q
% function. The Q function is expected to return optimal policy table based
% on an e-greedy policy.
%%
clear
clc
global Q_table
my_data=[2019,2019,2559,2559,7159,7159,8059,8059;0.1437,0.1533,0.1437,0.1533,0.1437,
0.1533,0.1437,0.1533];
% Define training Data per control level
env=my_data;
% Send Data to workspace for default startup
% date_str=24*(my_data(:,1));
% Define the Paramaters for Q-learning Algorithm
action_space=action_selection();
nreport=5;
lr=0.1;
num_episodes=100;
eps=0.4;
gama=0.95;
decay_eps=0.001;
[state_space,~]=size(env');
% Print Learning Message
fprintf('%s\n','Q-Learning function called')
fprintf('%s\n','Agent Learning the Environment...')
% Optimal Q-table
[Q_table,total_rew,set_val]=Q_learningfin(env,gama,num_episodes,eps,lr,decay_eps,action_
space,state_space);
```

```
% Generate Action binary form
%%
% Conversion into kW for easier assessment
Q_table=Q_table*10^-3;
oper_vec_man=0;

%%
```

Sub-function

Q-learning sub-function

```
function
[ret_opt,total_rew,set_rew]=Q_learningfin(env,gm,num_eps,eps,learn,eps_dec,action_space,
no_of_state)
% Define the Number of State and Action Space
% Define Q(s,a) to zeros
Q_sa=zeros(no_of_state,action_space);
mod_int=20;
total_rew=zeros(1,num_eps/mod_int);
set_rew=zeros(1,num_eps/mod_int);
count2=0;
for k=1:num_eps % Iteration # per episode
    count=0;
    state=1;
    while (1)
    count=count+1;
    if eps>0.1
        eps=eps-eps_dec;
    end
     action=eps_greedy(Q_sa,state,eps);
     next_state=randperm(no_of_state,1);
     env(:,state);
     rew_vect=my_fis(env(:,state));
```

```
    rew=rew_vect(action);
    Q_sa(state,action)=Q_sa(state,action)+learn*(rew+gm*max(Q_sa(next_state,:))-
Q_sa(state,action));
    state=next_state;
    % tot_rew=rew;
      if count>50
        break;
      end
   end
   % Call the Episodes function at intervals of 20
    % Q - learning curve
   if mod(k,mod_int)==0
      % Call the episodes function
      count2=count2+1;
      total_rew(count2)=run_epi(Q_sa,env);
      set_rew(count2)=k;
   end

ret_opt=Q_sa;
end
```

ϵ-greedy sub-function

```
% Action-value selection with greedy method
   function ret=eps_greedy(Q,index_s,eps)
      % Generate random number between 0 and 1
      rand_num=rand(1,1);
      [~,col]=size(Q);
      % Check if the rand_num is less than eps
      if rand_num<eps % Perform random selection
         ret=randperm(col,1);
      else
         [~,ind]=max(Q(index_s,:));
         ret=ind;
```

```
        end
    end
```

Arduino Implementation sub-function

```
% Reset the Arduino Board
clc
serx=serial('COM19','BAUD', 9600);
% my_a=arduino();
fopen(serx);
% my_a=arduino();
act_ard=zeros(1,26);
hbool = 1;
act_ard(1,3:26)=act_bin_over;
[~,rowd]=size(act_ard);
action_loadr=load_curve(:,4);


while hbool<=rowd
    if hbool<3
        fprintf('%s\t%d\n','Set-up : ',hbool);
    else
        fprintf('%s\t%d\n','Time : ',hbool-3);
    end
if (act_ard(hbool)==1)
    chk=load_curve(hbool,4);
    if (chk>=0.25 && chk<0.317)
        fprintf(serx,5*act_ard(hbool));
    elseif chk>0.317 && chk<1.88
        fprintf(serx,6*act_ard(hbool));
    elseif (chk>=1.8883 && chk<1.9)
        fprintf(serx,7*act_ard(hbool));
    end
else
    fprintf(serx,act_ard(hbool));
```

```
end
pause(20);
hbool=hbool+1;
end
```

Reset sub-function

```
%%
% Reset the Arduino Board
if ~isempty(instrfind)
   fclose(instrfind);
   delete(instrfind);
end
clc
```

Display sub-function

```
% This sub function implements the optimal policy at each state
% It also plots figures and charts
%%
% Derive the recommended action in binary form
act_bin=zeros(state_space,action_space);
%
for k=1:state_space
   [~,col]=max(Q_table(k,:));
   act_bin(k,col)=1;
end
% Determine the state of current load per load demand and tariff
% Read Data with load
control_load=[2019,2559,7159,8059];
tariff=0.1377;
load_ratio=control_load/max(control_load);
filename='app_data.xlsx';
filename2='tariff.xlsx';
```

```
sheet=1;
xlRange='A1:D24';
xlRange2='A1:B28';
load_curve=xlsread(filename,sheet,xlRange);
tar_data=xlsread(filename2,sheet,xlRange2);
load_mod=load_curve;
discount_fact=0.9;
[row,~]=size(load_curve);
load_patt=zeros(row,1);
load_clip=zeros(row,1);
load_valley=zeros(row,1);
load_sq=zeros(row,1);
max_load=max(load_curve(:,2))/discount_fact;
global act_bin_over;
act_bin_over=zeros(row,1);
ard_bin_over=zeros(row,1);
% Locate points of low tariff
tar_low=0.1438;
tar_low_vec=zeros(row,1);

for k=1:row
   % Define state
   % Locate tariff points
   if load_curve(k,4)*(8059)<=control_load(1) && load_curve(k,3)<=tariff
      % State One achieve
      [~,col]=max(act_bin(1,:));
      act_bin_over(k)=col;
   elseif load_curve(k,4)*8059<=control_load(1) && load_curve(k,3)>tariff
     [~,col]=max(act_bin(2,:));
      act_bin_over(k)=col;
   elseif load_curve(k,4)*8059>control_load(1) &&
load_curve(k,4)*(8059)<=control_load(2) && load_curve(k,3)<=tariff
      [~,col]=max(act_bin(3,:));
```

```
        act_bin_over(k)=col;
    elseif load_curve(k,4)*8059>control_load(1) &&
load_curve(k,4)*(8059)<=control_load(2) && load_curve(k,3)>tariff
        [~,col]=max(act_bin(4,:));
        act_bin_over(k)=col;
    elseif load_curve(k,4)*8059>control_load(2) &&
load_curve(k,4)*(8059)<=control_load(3) && load_curve(k,3)<=tariff
        [~,col]=max(act_bin(5,:));
        act_bin_over(k)=col;
    elseif load_curve(k,4)*8059>control_load(2) &&
load_curve(k,4)*(8059)<=control_load(3) && load_curve(k,3)>tariff
        [~,col]=max(act_bin(6,:));
        act_bin_over(k)=col;
    elseif load_curve(k,4)*8059>control_load(3) &&
load_curve(k,4)*(8059)<=control_load(4) && load_curve(k,3)<=tariff
        [~,col]=max(act_bin(7,:));
        act_bin_over(k)=col;
  elseif load_curve(k,4)*8059>control_load(3) && load_curve(k,4)*(8059)<=control_load(4)
&& load_curve(k,3)>tariff
        [~,col]=max(act_bin(8,:));
        act_bin_over(k)=col;
    end
end

% Return sum of loads to be shifted
imp=sum(tar_low_vec);
sum_vf=zeros(imp,1);
[row,col]=size(load_curve);
iter=1;
iter1=1;
load_dist=0;
cut=control_load.*load_ratio;
m=0;
```

```
sum_no_v=zeros(row,1);
for k=1:row
    % Compare loads
    % Check point where the is one
        if act_bin_over(k)==4
        tar_low_vec(k)=1;
        m=m+1;
        sum_vf(m)=load_curve(k,2);
        end
        % Region of low tariff
    %
    h=act_bin_over(k);
    switch h
        case 1
            % SQ State One achieved : Action Valley appliance with load IN1
           % Valley fill acording to ratio of loads

        case 2 % Load Shifting
            %load_mod(k,2)=(0.57+0.11)*max_load/6;
            if load_curve(k,2)>=cut(2) && load_curve(k,2)<cut(3)
                load_patt(k)=0.11*max_load;
                load_mod(k,2)=load_mod(k,2)-load_patt(k);
            elseif load_curve(k,2)>=cut(3)
                load_clip(k)=0.57*max_load;
                load_mod(k,2)=load_mod(k,2)-load_clip(k);
             end
            % load_dist=load_dist+(0.57+0.11)*max_load/6;
        case 3
         % LC State Three achieved : Action Valley fill with appliance IN2
         % load_mod(k,2)=load_mod(k,2)-0.11*max_load;
         if load_curve(k,2)>cut(3)
                load_clip(k)=0.57*max_load;
                load_mod(k,2)=load_mod(k,2)-load_clip(k);
```

```
        end

       case 4
       % State Four achieved
       % Appliances cant be interupted, so retain no action
    end

end
% Load distribution by load weight
count=0;
sm_def=sum(sum_no_v);
for k=1:row
    % Compare loads
    if load_curve(k,3)<=0.1377
            sum_no_v(k)=load_curve(k,2);
    end

    h=act_bin_over(k);
    switch h
       case 1
          % SQ State One achieved : Action Valley appliance with load IN1
        % Valley fill acording to ratio of loads
       case 2

       case 3

       case 4
       % State Four Valley fill based on ratio
       d=(load_mod(k,2)/sum(sum_vf))*(sum(load_patt)+sum(load_clip));
       load_mod(k,2)=load_mod(k,2)+d;
       load_valley(k)=d;
       count=1;
```

```
    end
end
% Allocate if there is no case four by default to low tariff region
if count==0
    % Allocate to regin of low
    for k=1:row

        d=(sum_no_v(k)/sum(sum_no_v))*(sum(load_patt)+sum(load_clip));
        load_mod(k,2)=load_mod(k,2)+d;
        load_valley(k)=d;
        if sum_no_v(k)>0
            act_bin_over(k)=4;
        end
        count=1;
    end
end
% Electricity Cost
energy_cost=zeros(row,2);
energy_cost(:,1)=load_curve(:,2).*load_curve(:,3)*power(1000,-1);
energy_cost(:,2)=load_curve(:,3).*load_mod(:,2)*power(1000,-1);
cost_comb=[energy_cost(:,1),energy_cost(:,2)];
energ_sav=zeros(row,1);
energy_save=(energy_cost(:,1)-energy_cost(:,2));
zer_v=(energy_save(:,1)<0);
non_zer_v=(energy_save(:,1)>0);
% Determine points of negative and positive savings
dh=energy_cost(:,1)-energy_cost(:,2);
col_dhn=dh(:,1)>0;% Paying Less and saving
col_dh=col_dhn;% col_dhp;
col_dhp=dh(:,1)<0; % Paying more
% Sum energy where DR effect took place
%%
% Return load_sq
```

```
load_sq_v=(load_curve(:,2)-load_mod(:,2)==0);
% Prepare minimum and maximum points
xmin=min(24*load_curve(:,1))-1;
xmax=1+max(24*load_curve(:,1));
ymin=0;
ymax=ceil(max(load_curve(:,2)));
ymax1=ceil(max(load_patt));
ymax3=ceil(max(load_valley));

bar_stack=zeros(row,3);
% Total Load
acc_rate=0.84;
hh=1506888; % Number of Households in Nairobi
partn=0.4; % 50% of the population contribution to DR Program
eff_hh=hh*acc_rate*partn*load_patt*power(10,-6);
 % Figures
figure(1)
set(gcf,'color','w');
subplot(2,2,1)
hold on
yyaxis left
bar(24*load_curve(:,1),load_curve(:,2));
grid('minor')
xlabel('Time (hrs)')
ylabel('Load (W)')
axis([xmin xmax ymin ymax])
yyaxis right
p=plot(24*tar_data(:,1),tar_data(:,2));
p.LineWidth = 3;
% p.FaceColor = [ 0 0.447 0.741];
ylabel('Tariff (US$/kWh)')
lgd=legend('Load Curve','Tariff','Orientation','horizontal','Location','northoutside');
title(lgd,'Initial Load Curve')
```

```
lgd.FontSize = 9;
hold off

subplot(2,2,2)
hold on
yyaxis left
bar(24*load_curve(:,1),load_mod(:,2));
grid('minor')
xlabel('Time (hrs)')
ylabel('Load (W)')
axis([xmin xmax ymin ymax])
yyaxis right
p=plot(24*tar_data(:,1),tar_data(:,2));
p.LineWidth = 3;
% p.FaceColor = [ 0 0.447 0.741];
ylabel('Tariff (US$/kWh)')
lgd=legend('Load Curve','Tariff','Orientation','horizontal','Location','northoutside');
title(lgd,'Modified Load Curve')
lgd.FontSize = 9;
hold off

% Status Quo
subplot(2,2,3)
hold on
yyaxis left
bar(24*load_curve(load_sq_v,1),load_curve(load_sq_v,2));
grid('minor')
xlabel('Time (hrs)')
ylabel('Load (W)')
ymax=ceil(max(load_curve(load_sq_v,2)));
axis([xmin xmax ymin ymax])
yyaxis right
p=plot(24*tar_data(:,1),tar_data(:,2));
```

```
p.LineWidth = 3;
% b.FaceColor = [ 0 0.447 0.741];
ylabel('Tariff (US$/kWh)')
lgd=legend('Load Curve','Tariff','Orientation','horizontal','Location','northoutside');
title(lgd,'Action - Status Quo')
lgd.FontSize = 9;
hold off

% Load Clipping
subplot(2,2,4)
hold on
yyaxis left
bar(24*load_curve(:,1),-1*load_clip,'r');
grid('minor')
xlabel('Time (hrs)')
ylabel('Load (W)')
ymax=ceil(max(load_patt));
axis([xmin xmax -1*ymax ymin])
yyaxis right
p=plot(24*tar_data(:,1),tar_data(:,2));
p.LineWidth = 3;
% b.FaceColor = [ 0 0.447 0.741];
ylabel('Tariff (US$/kWh)')
lgd=legend('Load Curve','Tariff','Orientation','horizontal','Location','northoutside');
title(lgd,'Action - Load Clipping')
lgd.FontSize = 9;
hold off

figure(2)
set(gcf,'color','w');
subplot(2,2,1)
hold on
yyaxis left
```

```
bar(24*load_curve(:,1),-1*load_patt);
grid('minor')
xlabel('Time (hrs)')
ylabel('Load (W)')
ymax=ceil(max(load_patt));
axis([xmin xmax -1*ymax ymin])
yyaxis right
p=plot(24*tar_data(:,1),tar_data(:,2));
p.LineWidth = 3;
% b.FaceColor = [ 0 0.447 0.741];
ylabel('Tariff (US$/kWh)')
lgd=legend('Load Curve','Tariff','Orientation','horizontal','Location','northoutside');
title(lgd,'Action - Load Shifting')
lgd.FontSize = 9;
hold off

subplot(2,2,2)
hold on
yyaxis left
bar(24*load_curve(:,1),load_valley,'g');
grid('minor')
xlabel('Time (hrs)')
ylabel('Load (W)')
axis([xmin xmax ymin ymax3])
yyaxis right
p=plot(24*tar_data(:,1),tar_data(:,2));
p.LineWidth = 3;
% p.FaceColor = [ 0 0.447 0.741];
ylabel('Tariff (US$/kWh)')
lgd=legend('Load Curve','Tariff','Orientation','horizontal','Location','northoutside');
title(lgd,'Action - Valley Filling')
lgd.FontSize = 9;
hold off
```

```
subplot(2,2,3)
hold on
grid('minor')
xlabel('Time (hrs)')
yt1=max(energy_cost(:,1));
yt2=max(energy_cost(:,2));
if yt1>yt2
   ymax=yt1;
else
   ymax=yt2;
end
axis([xmin xmax ymin ymax])
b=bar(24*load_curve(:,1),cost_comb,'grouped','g');
b(1).FaceColor=[1 0.2 0.2];
b(2).FaceColor=[0 0.6 0.3];
% plot(load_curve(:,1),energy_cost(:,2))
ylabel('Cost (US$)')
lgd=legend('Cost Before DR','Cost After
DR','Orientation','horizontal','Location','northoutside');
title(lgd,'Electricity Cost')
lgd.FontSize = 9;
hold off

subplot(2,2,4)
hold on
grid('minor')
xlabel('Time (hrs)')
bar(24*load_curve(non_zer_v,1),energy_save(non_zer_v),'g')
bar(24*load_curve(zer_v,1),energy_save(zer_v),'r')
%legend('Energy
% plot(load_curve(:,1),energy_cost(:,2))
ylabel('Cost \delta(US$)')
```

```
lgd=legend('Energy Savings due to DR','Energy
Expenses','Orientation','horizontal','Location','northoutside');
title(lgd,'Net Energy Cost Due to DR')
lgd.FontSize = 9;
hold off

figure(3)
set(gcf,'color','w');
% set(0,'defaultfigurecolor',[1 1 1])
subplot(1,2,1)
hold on
grid on
change_dh=sum(dh(col_dh==1,1))/(sum(energy_cost(col_dh==1,1))+sum(energy_cost(col_d
hp==1,1)));
my_pie=[1-change_dh,change_dh]*100;
axis([-2 2 -2 2])
h=pie(my_pie);
patchHand = findobj(h, 'Type', 'Patch');
patchHand(1).FaceColor = 'r';
patchHand(2).FaceColor = 'g';
lgd=legend('Energy Expenses','Savings due to
DR','Orientation','horizontal','Location','northoutside');
title(lgd,'DR Cost Savings Effect (Load Shifting and Clipping)')
lgd.FontSize = 9;
hold off

% Energy Save
subplot(1,2,2)
hold on
plot(set_val,total_rew,'LineWidth',2,'Color','r');
grid on
ylabel('Average Test Rewards')
xlabel('Training Time (hours)/Episodes')
```

```
lgd=legend('Agent Q-Learning Curve','Orientation','horizontal','Location','northoutside');
title(lgd,'Q-Learning Curve')
lgd.FontSize = 9;
hold off
% % Implement the Arduino Board
% Call the Arduino Matlab Serial Function

figure(4)
set(gcf,'color','w');
subplot(2,2,1)
hold on
bar(24*load_curve(:,1),eff_hh);
grid('minor')
axis([-1 24 0 max(eff_hh)*1.1])
xlabel('Time (hrs)')
ylabel('Load (MW)')
lgd=legend('Load Curve','Orientation','horizontal','Location','northoutside');
title(lgd,'Overall Effect of Load Shifting in Nairobi County')
lgd.FontSize = 9;
hold off

subplot(2,2,2)
hold on
grid on
change_dhp=-
0.8*sum(dh(col_dhp==1,1))/(sum(energy_cost(col_dh==1,1))+sum(energy_cost(col_dhp==1
,1)));
gh=dh/sum(energy_cost(:,2));
my_pie=[1-change_dhp,change_dhp]*100;
axis([-2 2 -2 2])
explode=[1 1];
h=pie(my_pie);
% newColors = [1,0.41016,0.70313;0,1,0.49609];
```

```
patchHand = findobj(h, 'Type', 'Patch');
% set(patchHand, {'FaceColor'}, mat2cell(newColors, ones(size(newColors,1),1), 3))
patchHand(1).FaceColor = 'b';
patchHand(2).FaceColor = 'r';
lgd=legend('Initial Energy Expense','Additional Energy Expense due to
DR','Orientation','horizontal','Location','northoutside');
title(lgd,'DR Energy Expense Effect (Valley Filling)')
lgd.FontSize = 9;
hold off

subplot(2,2,3)
hold on
grid on
% p=plot(24*tar_data(:,1),tar_data(:,2),'r','LineWidth',2);
% bar(24*tar_data(:,1),tar_data(:,2));
myColors=[0.4660, 0.6740, 0.1880;0, 0.75, 0.75;0.8500, 0.3250, 0.0980];
h=area(24*tar_data(1:7,1),tar_data(1:7,2));
h.FaceColor=myColors(1,:);
h=area(24*tar_data(8:15,1),tar_data(8:15,2));
h.FaceColor=myColors(2,:);
h=area(24*tar_data(16:22,1),tar_data(16:22,2));
h.FaceColor=myColors(3,:);
h=area(24*tar_data(23:25,1),tar_data(23:25,2));
h.FaceColor=myColors(2,:);
h=area(24*tar_data(26:28,1),tar_data(26:28,2));
h.FaceColor=myColors(1,:);
axis([0 24 0.135 0.155])
lgd=legend('Off-Peak','Mid-Peak','Peak','Orientation','horizontal','Location','northoutside');
title(lgd,'Time of Use Tariff Structure')
xlabel('Time (hrs)')
ylabel('US$/kWh')
lgd.FontSize = 9;
hold off
```

```
subplot(2,2,4)
set(gcf,'color','w');
hold on
yyaxis left
bar_stack_pos=[load_curve(:,2),load_valley];
bar_stack_neg=-1*(load_patt+load_clip);
b=bar(24*load_curve(:,1)',bar_stack_pos,'stacked');
%b(1).FaceColor=[0, 0.4470, 0.7410];
b(2).FaceColor=[0, 0.5, 0];
b=bar(24*load_curve(:,1)',bar_stack_neg,'stacked');
b(1).FaceColor=[1, 0, 0];

action_bin=0;

grid('minor')
xlabel('Time (hrs)')
ylabel('Load (W)')
yyaxis right
p=plot(24*tar_data(:,1),tar_data(:,2));
p.LineWidth = 3;
% p.FaceColor = [ 0 0.447 0.741];
ylabel('Tariff (US$/kWh)')
%axis([xmin xmax ymin ymax])
% p.FaceColor = [ 0 0.447 0.741];
ylabel('Tariff (US$/kWh)')
lgd=legend('Initial Load Curve/Status Quo','Valley Filling','Load Clipping &
Shifting','Orientation','horizontal','Location','northoutside');
title(lgd,'Overlay of Load Management Strategies')
lgd.FontSize = 9;
hold off
```

Arduino Computer Program

Implementation function

Preloaded function

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);
int value;
String my_time;
int relay[]={2,3,4,5};
int oper=10000;
int k=0;
int h=0;

void setup()
{
  // Start Diplay I2C LCD
  lcd.begin();
  lcd.backlight();
  Serial.begin(9600);
  // Set PINS 2, 3,4 5 as I/O
  pinMode(2,OUTPUT);
  pinMode(3,OUTPUT);
  pinMode(4,OUTPUT);
  pinMode(5,OUTPUT);
  // LC 1 PIN 2
  // LC 2 PIN 3
  // LC 3 PIN 4
  // LC 4 PIN 5
}
void loop()
```

```
{
 // Status Quo - Implies Checking channels that are closed and maintaining current flow.
 if(value>=1 && value<=1.89)
 {
   //Initialize Display
   lcd.clear();
   lcd.print(String(h) + String(" : 00 HRS"));
   lcd.setCursor (0,1); // go to start of 2nd line
   lcd.print("STATUS QUO");

  if(value-1>=0.25 && value-1<0.26){
   // Maintain Load Control 1
   digitalWrite(relay[0],LOW);
   delay(oper);
   digitalWrite(relay[0],HIGH);
  }

  else if(value-1>0.2503 && value-1<0.319){
   //Maintain Load Control 2
   digitalWrite(relay[0],LOW);
   digitalWrite(relay[1],LOW);
   delay(oper);
   digitalWrite(relay[0],HIGH);
   digitalWrite(relay[1],HIGH);
  }
  else if(value-1>0.3175 && value-1<0.89){
   // Maintain Load Control 3
   digitalWrite(relay[0],LOW);
   digitalWrite(relay[1],LOW);
   digitalWrite(relay[2],LOW);
   delay(oper);
   digitalWrite(relay[0],HIGH);
   digitalWrite(relay[1],HIGH);
```

```
    digitalWrite(relay[2],HIGH);
  }
  h++;
}
//Load shifting Means swithing off certan appliances -  Category 3 and 4.
else if (value==2)
{
        lcd.clear();
        lcd.print(String(h) + String(" : 00 HRS"));
        lcd.setCursor (0,1);
        lcd.print("LOAD SHIFTING");
        //Turn on relay
        digitalWrite(relay[0],LOW);
        digitalWrite(relay[1],LOW);
        digitalWrite(relay[2],HIGH);
        digitalWrite(relay[3],HIGH);
        delay(oper);// Delay by operation time (HRS)
        // Turn off relay
        digitalWrite(relay[0],HIGH);
        digitalWrite(relay[1],HIGH);
        h++;
}

// Load clipping Means Turning off appliances in category 4.
else if (value==3)
{
        lcd.clear();
        lcd.print(String(h) + String(" : 00 HRS"));
        lcd.setCursor (0,1);
        lcd.print("LOAD CLIPPING");
        // Turn on Relay
        digitalWrite(relay[0],LOW);
        digitalWrite(relay[1],LOW);
```

```
        digitalWrite(relay[2],LOW);
        delay(oper);// Delay by operation time (HRS)
        // Turn off relay
        digitalWrite(relay[0],HIGH);
        digitalWrite(relay[1],HIGH);
        digitalWrite(relay[2],HIGH);
        digitalWrite(relay[3],HIGH);
        h++;
 }
  // Valley Filling means turning on appliances 3 or 4.
else if (value==4)
 {
        lcd.clear();
        lcd.print(String(h) + String(" : 00 HRS"));
        lcd.setCursor (0,1);
        lcd.print("VALLEY FILLING");
        // Turn on relay
        digitalWrite(relay[0],LOW);
        digitalWrite(relay[1],LOW);
        digitalWrite(relay[2],LOW);
        digitalWrite(relay[3],LOW);
        delay(oper);// Delay by operation time (HRS)
        // Turn off relay
        digitalWrite(relay[0],HIGH);
        digitalWrite(relay[1],HIGH);
        digitalWrite(relay[2],HIGH);
        digitalWrite(relay[3],HIGH);
        h++;
 }
else
 {
        lcd.clear();
        lcd.print("WAITING FOR ");
```

```
        lcd.setCursor (0,1);
        lcd.print("OPTIMAL POLICY");
        delay(1000);// Delay by operation time (HRS)
        // Turn off relay
        digitalWrite(relay[0],HIGH);
        digitalWrite(relay[1],HIGH);
        digitalWrite(relay[2],HIGH);
        digitalWrite(relay[3],HIGH);

  }
        digitalWrite(relay[0],HIGH);
        digitalWrite(relay[1],HIGH);
        digitalWrite(relay[2],HIGH);
        digitalWrite(relay[3],HIGH);

  }
```

Current Measurement Function

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);
// Initialize variables
int value_vec[]={4,4,4,4,4,4,4,2,2,2,2,2,2,2,2,2,6,6,2,2,2,2,4,4};
int value;
String my_time;
int relay[]={2,3,4,5};
int oper=10000;
int k=0;
int h=0;

int mVperAmp = 100;
```

```
float mysum=0;
double Voltage = 0;
double VRMS = 0;
double AmpsRMS=0;
float readValue[]={0,0,0,0};
void setup()
{

  // Start Diplay I2C LCD
  lcd.begin();
  lcd.backlight();
  Serial.begin(9600);
  // Set PINS 2, 3,4 5 as I/O
  pinMode(2,OUTPUT);
  pinMode(3,OUTPUT);
  pinMode(4,OUTPUT);
  pinMode(5,OUTPUT);
  // LC 1 PIN 2
  // LC 2 PIN 3
  // LC 3 PIN 4
  // LC 4 PIN 5
}

void loop()
{
for (int q=0;q<24;q++){
  value=value_vec[q];
  mysum=0;
  // Status Quo - Implies Checking channels that are closed and maintaining current flow.
  if(value>=5 && value<=7)
  {
```

```
    //Initialize Display
    lcd.clear();
    lcd.print(String(h) + String(" : 00 HRS"));
    lcd.setCursor (0,1); // go to start of 2nd line
    lcd.print("STATUS QUO");

  if(value==5){
    // Maintain Load Control 1
    digitalWrite(relay[0],LOW);
    mysum = (((analogRead(A0)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A1)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A2)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A3)) * 5.0)/1024.0-2.5+0.122070313)/0.1;
    delay(oper);
    digitalWrite(relay[0],HIGH);
  }

  else if(value==6){
    //Maintain Load Control 2
    digitalWrite(relay[0],LOW);
    digitalWrite(relay[1],LOW);
    mysum = (((analogRead(A0)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A1)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A2)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A3)) * 5.0)/1024.0-2.5+0.122070313)/0.1;
    delay(oper);
    digitalWrite(relay[0],HIGH);
    digitalWrite(relay[1],HIGH);
  }

  else if(value==7){
    // Maintain Load Control 3
    digitalWrite(relay[0],LOW);
    digitalWrite(relay[1],LOW);
```

```
      digitalWrite(relay[2],LOW);
      mysum = (((analogRead(A0)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A1)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A2)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A3)) * 5.0)/1024.0-2.5+0.122070313)/0.1;
      delay(oper);
      digitalWrite(relay[0],HIGH);
      digitalWrite(relay[1],HIGH);
      digitalWrite(relay[2],HIGH);
    }
    h++;
  }
  //Load shifting Means swithing off certan appliances -  Category 3 and 4.
  else if (value==2)
  {
        lcd.clear();
        lcd.print(String(h) + String(" : 00 HRS"));
        lcd.setCursor (0,1);
        lcd.print("LOAD SHIFTING");
        //Turn on relay
        digitalWrite(relay[0],LOW);
        digitalWrite(relay[1],LOW);
        digitalWrite(relay[2],HIGH);
        digitalWrite(relay[3],HIGH);
        mysum = (((analogRead(A0)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A1)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A2)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A3)) * 5.0)/1024.0-2.5+0.122070313)/0.1;
        delay(oper);// Delay by operation time (HRS)
        // Turn off relay
        digitalWrite(relay[0],HIGH);
        digitalWrite(relay[1],HIGH);
        h++;
  }
```

```
// Load clipping Means Turning off appliances in category 4.
else if (value==3)
{
      lcd.clear();
      lcd.print(String(h) + String(" : 00 HRS"));
      lcd.setCursor (0,1);
      lcd.print("LOAD CLIPPING");
      // Turn on Relay
      digitalWrite(relay[0],LOW);
      digitalWrite(relay[1],LOW);
      digitalWrite(relay[2],LOW);
      mysum = (((analogRead(A0)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A1)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A2)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A3)) * 5.0)/1024.0-2.5+0.122070313)/0.1;

    delay(oper);// Delay by operation time (HRS)
      // Turn off relay
      digitalWrite(relay[0],HIGH);
      digitalWrite(relay[1],HIGH);
      digitalWrite(relay[2],HIGH);
      digitalWrite(relay[3],HIGH);
      h++;
}
  // Valley Filling means turning on appliances 3 or 4.
else if (value==4)
{
      lcd.clear();
      lcd.print(String(h) + String(" : 00 HRS"));
      lcd.setCursor (0,1);
      lcd.print("VALLEY FILLING");
      // Turn on relay
```

```
        digitalWrite(relay[0],LOW);
        digitalWrite(relay[1],LOW);
        digitalWrite(relay[2],LOW);
        digitalWrite(relay[3],LOW);
        mysum = (((analogRead(A0)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A1)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A2)) * 5.0)/1024.0-2.5+0.122070313)/0.1+(((analogRead(A3)) * 5.0)/1024.0-2.5+0.122070313)/0.1;
        delay(oper);// Delay by operation time (HRS)
        // Turn off relay
        digitalWrite(relay[0],HIGH);
        digitalWrite(relay[1],HIGH);
        digitalWrite(relay[2],HIGH);
        digitalWrite(relay[3],HIGH);
        h++;
 }

 else
 {
        lcd.clear();
        lcd.print("WAITING FOR ");
        lcd.setCursor (0,1);
        lcd.print("OPTIMAL POLICY");
        delay(1000);// Delay by operation time (HRS)
        // Turn off relay
        digitalWrite(relay[0],HIGH);
        digitalWrite(relay[1],HIGH);
        digitalWrite(relay[2],HIGH);
        digitalWrite(relay[3],HIGH);
        mysum=0;

 }
         Serial.println(mysum,3);
```

```
        digitalWrite(relay[0],HIGH);
        digitalWrite(relay[1],HIGH);
        digitalWrite(relay[2],HIGH);
        digitalWrite(relay[3],HIGH);

  }
  delay(1000);
}
```

Appendix VI: Questionnaire on Demand Response

Section 1 of 2

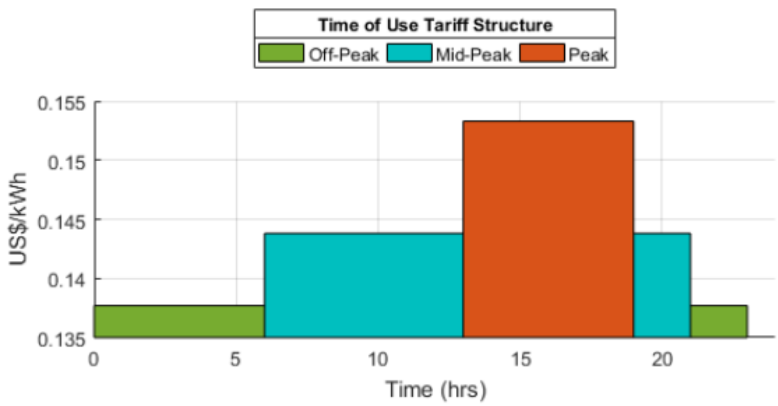
Short Survey on Time of Use (ToU) Tariff Plan - Residential Scale

Dear Sir/Madam,

My name is Walter Angano, a student researching demand response for a Smart Home Energy Management System (SHEMS). Part of the research involves obtaining feedback on the time of use tariff from residential energy consumers.
A time of use tariff enables you to get billed for energy consumption at varying tariffs depending on the Time of Use (ToU). During peak times (14:00:00 HRS - 20:00:00 HRS) the tariff is higher while lower during off-peak times (00:00:00 - 06:59:59 HRS). Moderate tariffs are offered during mid-peaks (07:00:00 - 13:59:59 HRS)

Thank You.
Walter Angano

Example of a Time of Use Tariff Structure



Title

This survey aims to gather your view on the topic and whether you as a consumer would be willing to accept the ToU tariff plan assuming it were to be introduced and what are some of the concerns you feel need to be addressed before its deployment.
Some of the benefits of Demand Response realized by other countries where ToU is active include:
a) Incentive payments
b) Energy Bill savings

Section 2 of 2

General Information

Description (optional)

Name of Respondent (optional)

Short answer text

Details of Respondent *

- Male
- Female
- Private

Have you heard about a time of use tariff? *

- Yes
- No

Electricity cost has been increasing and one of the possible ways to offset such costs is through demand-side management. Are you willing to accept a ToU plan? *

- Yes
- No
- Maybe

What percentage of energy savings would you prefer to participate in the ToU program. *

- 5%
- 10%
- 15%
- Greater than 20%

Are you willing to have a Smart Home Energy Management System on the basis that you will save energy and hence reduction in energy bills? *

- Yes
- No
- Maybe

Appendix VII: Research Paper Submission to IEEE PowerAfrica Conference 2021

Design and Testing of a Demand Response Q-Learning Algorithm for a Smart Home Energy Management System

Walter Angano
Department of Mechanical and Manufacturing Engineering
University of Nairobi
Nairobi, Kenya
walter.angano@students.uonbi.ac.ke

Peter Musau
Department of Electrical and Information Engineering
University of Nairobi
Nairobi, Kenya
pemosmusa@uonbi.ac.ke

Cyrus Wabuge Wekesa
School of Engineering
University of Eldoret
Eldoret, Kenya
cwekesa@uoeld.ac.ke

Abstract— **Growth in energy demand stimulates a need to meet this demand which is achieved either through wired solutions like investment in new or expansion of existing generation, transmission and distribution systems or non-wired solutions like Demand Response (DR). This paper proposes a Q-learning algorithm, an off-policy Reinforcement Learning technique, to implement DR in a residential energy system adopting a static Time of Use (ToU) tariff structure, reduce its learning speed by introducing a knowledge base that updates fuzzy logic rules based on consumer satisfaction feedback and minimize dissatisfaction error. Testing was done in a physical system by deploying the algorithm in Matlab and through serial communication interfacing the physical environment with the Arduino Uno. Load curve generated from appliances and ToU data was used to test the algorithm. The designed algorithm minimized electricity cost by 11 % and improved the learning speed of its agent within 500 episodes.**

Keywords— Demand Response, Q-Learning, Reinforcement Learning, Smart Home Energy Management System, Time of Use

I. INTRODUCTION

Electrical energy has the advantage of versatility (can be put to multiple uses), cleanliness and can be transported at the speed of light. However, one major problem this form of energy faces is the expense of providing grid-scale storage. For this reason, the energy generated must simultaneously be consumed. That is, energy generation must balance energy demand plus energy losses at all times, a necessity that also facilitates support for system integrity (constancy of system frequency). The Kenya Least Cost Power Development Plan (LCPDP) report findings forecast an excess generation compared to demand in the coming years [1] and the consequence is an increase in electricity prices to meet costs due to excess generation.

Demand Side Management (DSM) has been demonstrated as an effective tool for promoting energy efficiency and balance between energy generation and demand. DSM as an overarching topic encourages energy consumers and utilities to be energy efficient. The elements of DSM include Load Management and Demand Response (DR). As one of the vehicles of DSM, DR refers to short-term responses to electricity market prices on the demand side/ by consumers [2]. DR programs are developed to encourage short-term load reductions by consumers when the energy pricing is high particularly during peak hours. DR programs are categorized into price and incentive-based [2]. Examples of DR price-based programs include the Static Time of Use (ToU) rates, Critical peak pricing (CPP), and Real-Time Pricing (RTP). Research on DR algorithms has evolved with Q-Learning agent-based algorithm being the predominant method.

This paper proposes an approach objectively to decrease the learning speed of a Q-learning agent and integrating consumer feedback on optimal policy by an agent subject to a static ToU. The rest of the paper is organized as follows; Background and Related Work, Methodology, Results and Discussion, Conclusion, Acknowledgement and References.

II. BACKGROUND AND RELATED WORK

A review of DR algorithms and modeling techniques by [3] illustrates Reinforcement Learning (RL) as a predominantly applied method in DR applications when problems are formulated as a Markov Decision Process (MDP). RL algorithm is considered more suitable in real-world applications, particularly DR. One of the RL algorithms widely used in DR is Q-learning which is agent-environment-based and seeks to establish an optimal policy from a set of actions. The authors concluded that most reinforcement learning algorithms have been performed in a simulation environment which has limited the implementation of such algorithms in residential and commercial buildings. Testing of algorithms in physical systems is a potential research path to measure the capability, flexibility and reliability of control by reinforcement learning agents. Limited publications considered human feedback through estimation of dissatisfaction function. Some algorithms are characterized with a curse of dimensionality problem particularly for large state-action where the speed of convergence is significantly reduced and subsequently learning speed by RL agent.

Other approaches explored include intelligent residential consumer systems that trade with an Energy Storage System (ESS) while non-intelligent consumers are given the option of purchasing energy from the ESS pool [4]. Intelligent residential systems have a smart agent that manages the ESS based on the pool price and neighborhood energy demand. The authors preferred a fuzzy inference system for the battery and price by setting a fuzzy logic where values of input vector through fuzzy rules are translated into corresponding output vector. The fuzzy rules represent the infinite states of energy price and the State of Charge as finite states.

The authors [5] proposed a demand response scheme using RL with a single agent and integrates fuzzy reasoning to approximate values for reward functions. Human preference is considered in the control feedback as a state at each time step. Q-learning (an off-policy RL technique) was considered in selecting an optimal decision. The MDP constituted state-space with all the possible states in terms of power demand and electricity price signals. The reward function was implemented using fuzzy logic which approximates the numerical reward for a certain action and state. The actions with the highest reward values are considered optimal and corresponding actions implemented.

Multi-agent approaches included the design of a multi-agent RL intending to achieve an efficient home-based DR by modeling a one-hour ahead scheduling of smart appliances for a home energy management system with PV generation [6]. The proposed RL approach consists of two parts. The first part is a training of the Extreme Learning Machine (ELM) algorithm which is based on the feed-forward Neural Network. The ELM, using previous 24-hr data, predicts the 24-hr future trend on electricity prices and solar PV generation output. The predicted data is input to the second part which is a Q-learning algorithm designed to make hour-ahead decisions on energy consumption based on optimal policy. The optimal Q value is obtained using the Bellman equation. RL solution can be summarized to entail three algorithms, first algorithm the main function that initializes the parameters of the Q learning. The second algorithm is a feedforward NN with 24-hr data on electricity prices and solar generation as its input. The output is the predicted information on electricity price and solar generation for the next hour. The third algorithm is the Q-learning algorithm that makes scheduling decisions based on optimal policy.

Real-time DR was conducted to minimize the cost of electricity and maximize user comfort [7]. The authors presented an optimal scheduling strategy of appliances based on deep reinforcement learning (DRL) considering both discrete and continuous policies. An approximate policy was design based on the neural network (NN) to learn the optimal scheduling strategy from high-dimensional data of real-time pricing, states of an appliance, and outdoor temperature. The NN is trained using a policy search algorithm. The MDP structure consists of states as real-time electricity prices, outdoor temperature, and state of all appliances. Actions include binary control action variables/ discrete (deferrable appliances), continuous control variables (regulated appliances). Reward function modeled on three aspects: thermal comfort index, electricity cost, and consumer range anxiety. In solving the MDP, a neural network-based stochastic policy is adopted to determine the optimal policy. Bernoulli distribution and Gaussian distribution functions are used to estimate the approximate policy when the action is discrete and continuous, respectively. NN policy network determined the parameters for the distribution functions by learning them. The architecture of the NN takes in the input parameters (past electricity prices, outdoor temperatures, and states of all the appliances) and outputs the discrete and continuous actions by Bernoulli and Gaussian distribution functions respectively.

Most RL algorithms have been tested in simulation environments with limited testing in physical systems while others presented approaches that are considered complex for a simple residential system. In the context of integrating human feedback, the simulation environment limits actual feedback which is essential in understanding the performance of the agent. The curse of dimensionality has been addressed but learning speeds can still be significantly improved. Besides, the agent's action selection preference requires both exploitation and exploration of the environment. Multi-agent systems involved assigning an agent to each appliance which seems a complex system for small residential systems.

III. METHODOLOGY

A. Markov Decision Process (MDP) Model

1) Environment

The environment consists of non-schedulable appliances (mandatory) and schedulable (interruptible and non-interruptible) as the primary participants in DR. Schedulable appliances provide the leverage to deploy load management strategies per the ToU and realize energy savings. Load classification and load control level emanates from arranging load demand for the appliances according to consumer preference and priority and computing their cumulative load demand, respectively.

The total demand, P_t from all the appliances at any given time is given by equation

$$P_t = P_t^{NS} + P_t^{NI} + P_t^{IN1} + P_t^{IN2} \tag{1}$$

Load control levels (LCL) are defined cumulatively by adopting load demand for each appliance category.

$$LCL_1 = P^{NS} \tag{2}$$

$$LCL_2 = P^{NS} + P^{NI} \tag{3}$$

$$LCL_3 = P^{NS} + P^{NI} + P^{IN1} \tag{4}$$

$$LCL_4 = P^{NS} + P^{NI} + P^{IN1} + P^{IN2} \tag{5}$$

Where P^{NS}, P^{NI}, P^{IN1} and P^{IN2} is the total load demand for non-schedulable, non-interruptible, priority one and two interruptible appliances, respectively.

2) Agent

A single agent is designed and trained using data from a residential consumer and learns the environment for optimal policy output.

3) State Space

The set of state-space consists of the LCL and electricity static ToU.

4) Action Space

The action space consists of a set of load management strategies (load clipping, valley filling and load shifting) and status quo (no action). The balance in the action space is given in Fig 1. Load shifting and clipping actions are compensated by valley-filling.

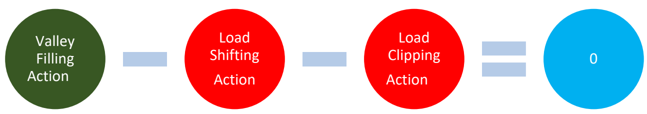


Fig. 1. Action space balance

The action space per the LCL and electricity price grid is assigned a weight which is essential when integrating feedback from consumers.

B. Reward Function

According to [8], the three main types of fuzzy logic systems commonly used include fuzzifier and defuzzifier, pure and Takagi-Segeno-Kang (TSK) fuzzy systems. A fuzzy system with fuzzifier and defuzzifier is commonly used as it eliminates problems associated with pure and TSK fuzzy systems. A Fuzzy logic system constitutes a crisp input (the LCL and static ToU) and crisp output is the numerical reward approximated by the system's fuzzy inference engine.

C. Fuzzy Rule Base and Fuzzy Inference Engine

A fuzzy rule base as the heart of the fuzzy system constitutes a set of IF-THEN rules. From Fig. 2, eight rules in a canonical form are defined in Table I. The fuzzy set **A** includes the LCL in a universe of discourse V equivalent to Load control 4 (LC4) and the status of electricity prices (whether Low or High) in a universe of discourse derived from historical tariff data.

TABLE I. THE CANONICAL FORM OF THE RULE BASE

Rule 1	If (LD is LCL1) and (ET is LP) then (SQ is NR)(LS is NR)(LC is NR)(VF is HR) (1)
Rule 2	If (LD is LCL1) and (ET is HP) then (SQ is HR)(LS is NR)(LC is NR)(VF is NR) (1)
Rule 3	If (LD is LCL2) and (ET is LP) then (SQ is LR)(LS is LR)(LC is NR)(VF is HR) (1)
Rule 4	If (LD is LCL2) and (ET is HP) then (SQ is HR)(LS is LR)(LC is NR)(VF is NR) (1)
Rule 5	If (LD is LCL3) and (ET is LP) then (SQ is HR)(LS is R)(LC is NR)(VF is NR) (1)
Rule 6	If (LD is LCL3) and (ET is HP) then (SQ is NR)(LS is HR)(LC is NR)(VF is NR) (1)
Rule 7	If (LD is LCL4) and (ET is LP) then (SQ is NR)(LS is NR)(LC is HR)(VF is NR) (1)
Rule 8	If (LD is LCL4) and (ET is HP) then (SQ is NR)(LS is NR)(LC is HR)(VF is NR) (1)

Fuzzy set B constitutes action space linguistic form Highly Recommended (HR), Recommended (R), Least Recommended (LR) and Not Recommended (NR).

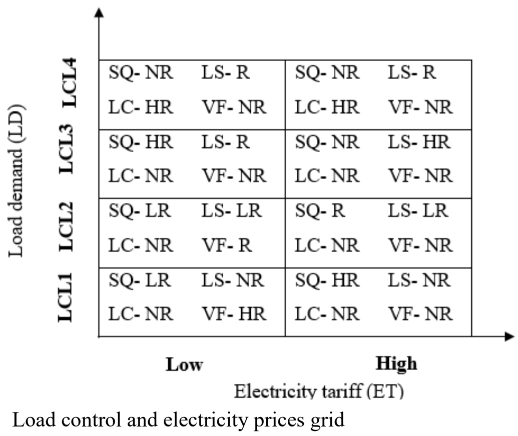


Fig. 2. Load control and electricity prices grid

Mamdani inference method, a type of composition-based inference, is adopted based on intuitive appeal. Mamdani combination is defined as a single fuzzy relation Q_M,

$$Q_M = \bigcup_{k=1}^{Y} Ru^k \tag{6}$$

A minimum inference engine is adopted in this research defined as,

$$\mu_{B^k}(y) = \max_{k=1}^{Y} \left[\sup_{x \in U} \min \begin{pmatrix} \mu_{A^k}(x), \mu_{A_1{}^k}(x_1), \ldots \\ , \mu_{A_n{}^k}(x_n), \mu_{B^K}(y) \end{pmatrix}\right] \tag{7}$$

Triangular fuzzifier maps a real value $\boldsymbol{x^{max}} \in \boldsymbol{U}$ to a fuzzy set $A^k \; in \, U$ characterized by a triangular membership function,

$$\mu_A(x) = \begin{cases} \left(1 - \dfrac{|x_1 - x_{max}|}{b_1}\right) \cdots \left(1 - \dfrac{|x_n - x_{max}|}{b_n}\right) \\ \quad if \; |x_i - x_{i,max}| \le b_i, i = 1,2,\ldots,n \\ 0 \qquad\qquad if \; otherwise \end{cases} \tag{8}$$

This paper adopts the center of gravity (CoG) defuzzifier. The CoG defuzzifier specifies y^* as the area center covered a membership of B^k as

$$y^* = \frac{\int_V y(\mu_{B^k})(y)dy}{\int_V (\mu_{B^k})(y)dy} \tag{9}$$

The defuzzifier in this case outputs the approximate reward based on the crisp input (LCL and static ToU).

D. Introduction to Q-Learning Algorithm

Q-learning algorithm is a temporal difference learning algorithm and an off-policy reinforcement learning that aims to learn optimal policy and approximates the current optimal action-value q_* using the Bellman equation,

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_t + \gamma max_a Q(s_{t+1}, a) - Q(s_t, a_t)\right] \tag{10}$$

Q-learning algorithm computes the value of taking an action a in state s and determines the optimal policy, $q_*(s, a)$ from a set of actions $a \in A(s)$ for that particular state. The parameters α and γ represent the learning rate of the algorithm and the discount factor [9], [10].

E. Exploration and Exploitation

This paper adopts the $\varepsilon - greedy$ exploration technique which ensures actions are selected randomly (exploration) and greedily (exploitation) with a probability ε and $1 - \varepsilon$, respectively.

$$a_t = \begin{cases} random\ action & with\ probability\ \varepsilon \\ maxQ_t(a) & with\ probability\ 1 - \varepsilon \end{cases} \tag{11}$$

F. Returns and Episodes

The primary goal of an agent is to maximize cumulative rewards in a particular time slot. Denote sequence of rewards as $\boldsymbol{R_{t+1}}, \boldsymbol{R_{t+2}}, \boldsymbol{R_{t+3}}, \ldots$ so that the expected return is maximized. The maximized return is considered a function of the sum of all rewards.

$$G_t = R_{t+1} + R_{t+2} + R_{t+3} + \cdots + R_T \tag{12}$$

Where T is the final time step or episode.

G. Integration of Consumer Feedback

Binary action vectors by the algorithm and consumer at time t are represented as $W_{Alg,t}$ and $W_{Cons,t}$, respectively. Then the magnitude of the vector can be used to determine consumer dissatisfaction. Consider the length of the action vector difference,

$$\Delta W = \| W_{Alg,t} - W_{Cons,t} \| \tag{13}$$

When, $\Delta W = 0$, the consumer is satisfied with the algorithm's optimal policy. However, the consumer shows dissatisfaction when $\Delta W > 0$. Consumer dissatisfaction with the algorithm's decision is handled by updating the fuzzy rules. However, the reward difference between consumer dissatisfaction and algorithm is minimized. The weighting method is used to assign the weights of the linguistic action space in Fuzzy set B as Highly Recommended (HR) – 0.4, Recommended (R) – 0.3, Least Recommended (LR)-0.2 and Not Recommended (NR) - 0.1. At *time t*, the algorithm's action vector and corresponding index are given as:

$$[\beta_{alg,t}, \lambda_{alg,t}] = max(W_{Alg,t}) \tag{14}$$

Consumer feedback is represented as

$$[\beta_{cons,t}, \lambda_{cons,t}] = max(W_{cons,t}) \tag{15}$$

The difference in reward needs to be greater than zero to guarantee an update to the rules. When $R_t(\lambda_{cons,t}) - R_t(\lambda_{alg,t}) > 0$, then the rules are updated depending on the weightage. The load demand and electricity price grid G is assigned the maximum of weighted Fuzzy set B.

$$G\left((ET_{n=1\ to\ 2}, LD_{m=1,\dots,4}), \lambda_{alg,t}\right) = G\left((ET_{n=1\ to\ 2}, LD_{m=1,\dots,4}), \lambda_{cons,t}\right) \tag{16}$$

$$G\left((ET_{n=1\ to\ 2}, LD_{m=1,\dots,4}), \lambda_{cons,t}\right) = \max(Fuzzy\ B) \tag{17}$$

An example of the fuzzy rule update is illustrated in Fig. 3.

SQ- 0.1	LS- 0.3	SQ- 0.1	LS- 0.3
LC- 0.4	VF- 0.1	LC- 0.4	VF- 0.1
SQ- 0.4	LS- 0.3	SQ- 0.1	LS- 0.4
LC- 0.1	VF- 0.1	LC- 0.1	VF- 0.1
SQ- 0.2	LS- 0.2	SQ- 0.3	LS- 0.2
LC- 0.1	VF- 0.3	LC- 0.1	VF- 0.1
SQ- 0.2	LS- 0.1	SQ- 0.4	LS- 0.1
LC- 0.1	VF- 0.4	LC- 0.1	VF- 0.1

→

SQ- 0.1	LS- 0.3	SQ- 0.1	LS- 0.3
LC- 0.4	VF- 0.1	LC- 0.4	VF- 0.1
SQ- **0.3**	LS- **0.1**	SQ- 0.1	LS- 0.4
LC- 0.1	VF- **0.4**	LC- 0.1	VF- 0.1
SQ- 0.2	LS- 0.2	SQ- 0.3	LS- 0.2
LC- 0.1	VF- 0.3	LC- 0.1	VF- 0.1
SQ- 0.2	LS- 0.1	SQ- 0.4	LS- 0.1
LC- 0.1	VF- 0.4	LC- 0.1	VF- 0.1

Fig. 3. Fuzzy rule update using the knowledge base

H. Time of Use Tariff Structure

The tariff structure is given in Fig.4. Historical tariff data for residential consumers in Kenya are distributed around the mean which is also the shoulder or mid-peak. ToU plans from Ireland Italy, Australia, Canada and Sri Lanka [12] are used in benchmarking.

I. Testing Setup

The testing set-up in Fig. 5 is implemented using the Arduino Uno kit. An Arduino program is preloaded in the microprocessor. This program checks if the serial port has changed for processing. Matlab program which is the agent communicates with the preloaded program through serial communication.

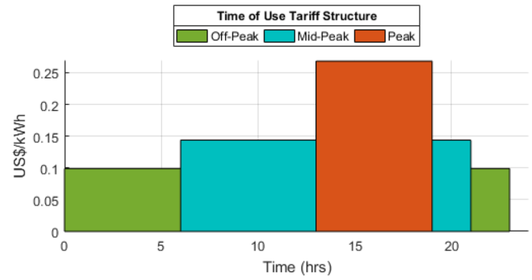


Fig. 4. Static Time of Use tariff plan

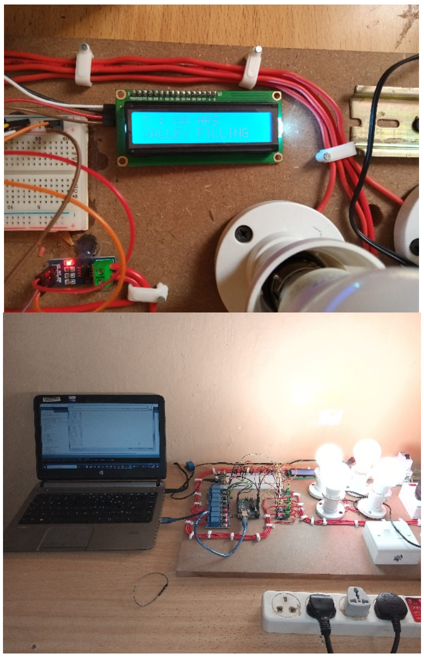


Fig. 5. Testing set-up for the algorithm

IV. Results and Discussion

Fig. 6 expresses the learning curve as the graph of mean cumulative rewards as a consequence of the agent's optimal policy selection against the number of episodes or training time taken. It was observed that the agent converged after 500 episodes.

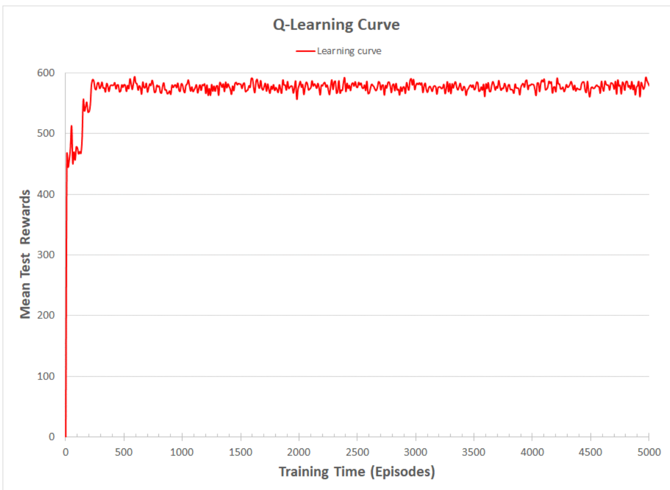


Fig. 6. The training curve for the algorithm

The overall effect of the agent's action is evaluated through an overlay of the recommended actions on the initial load curve. The applicable load management strategies include load shifting, valley-filling and status quo. Appliances in load category 3 were shifted during the peak times when the tariff is high. This happens between 07:00-16:00 and 18:00-21:00 HRS which are the mid-peak and peak hours

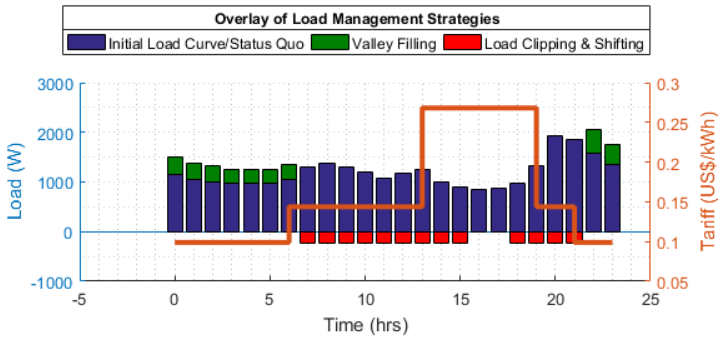


Fig. 7. Overlay of Load Management Strategies

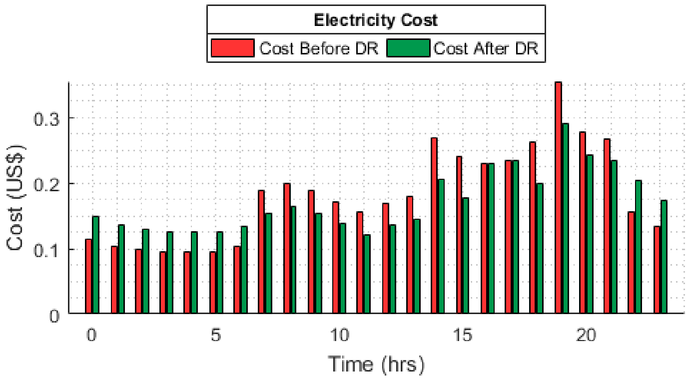


Fig. 8. Comparison of Energy Cost before and after demand response

For the status quo, the net cost is zero since no action is taken. Fig 7 shows that during valley filling, the demand response effect is an extra electricity cost on the consumer since appliances are added hence an increase in total load. The costs and savings as a result of the corresponding applicable load management strategies only at the region or time when they are applied as shown in Fig. 8. The net energy savings realized is 11 percent as in Fig. 9.

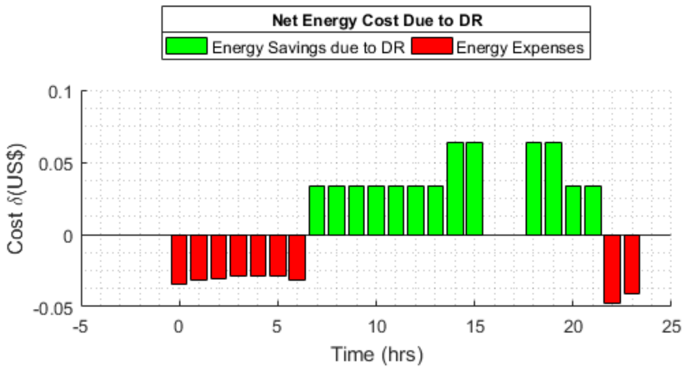


Fig. 9. Comparison of Energy Cost before and after demand response

V. Conclusion

Q-learning is a predominant tool of reinforcement learning that researchers have found resourceful when establishing optimal policy from a set of actions. This paper proposed improving some of the gaps by establishing a state-space action consisting of consumer-tailored load categories by grouping appliances according to their priority and usage frequency. A knowledge improvement base was developed to update the fuzzy rules and ensure the algorithm minimizes consumer dissatisfaction. This approach resulted in the agent's improved learning speed with convergence in 500 episodes and cost savings of 11 percent. A testing system was assembled and interfaced with a graphical user interface designer using app designer in Matlab. Through serial communication, the Arduino microprocessor received command signals from Matlab and either activated or deactivated a relay to turn the loads on or off.

Future research work will focus on developing consumer dissatisfaction models using Artificial Neural Network (ANN) and cloud-based technologies such as Microsoft Azure and integrating the models as a crisp input in fuzzy systems.

Acknowledgment

I acknowledge Virunga Power's support and encouragement during this research.

References

[1] Energy and Petroleum Regulatory Authority, "Updated Least Cost Development Plan Study Period 2020-2040," EPRA, Nairobi, 2020.

[2] United Nations Industrial Development Organization, "Sustainable Energy Regulation and Policy-making Training Manual - Demand Side Management," United Nations Industrial Development Organization, 2009.

[3] I. Hussain, S. Mohsin, A. Basit, Q. U. Khan ZA and N. Javaid, "A review on demand response: pricing, optimization, and appliance scheduling," *Procedia Computer Science,* vol. 52, p. 843–50, 2015.

[4] J. Vázquez-Canteli and Z. Nagy, "Reinforcement learning for demand response: A review of algorithms and modeling techniques," *Applied Energy,* vol. 235, pp. 1072-89, 2019.

[5] S. Zhou, Z. Hu, W. Gu, M. Jiang and X. Zhang, "Artificial intelligence based smart energy community management: A reinforcement learning approach," *CSEE Journal of Power and Energy Systems,* vol. 5, pp. 1-10, 2019.

[6] 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.

[7] X. Xu, Y. Jia, Y. Xu, Z. Xu, S. Chai and S. C. Lai, "IEEE Transactions on Smart Grid," *A Multi-Agent Reinforcement Learning-Based Data-Driven Method for Home Energy Management,* vol. 11, pp. 3201-3211, 2020.

[8] H. Li, Z. Wan and H. He, "Real-Time Residential Demand Response," *IEEE Transactions on Smart Grid,* vol. 11, pp. 4144-4154, 2020.

[9] L. X. Wang, A Course in Fuzzy Systems and Control, Prentice Hall PTR, 1997.

[10] R. S. Sutton and A. G. Barto, "Reinforcement Learning," in *Finite Markov Decision Processes*, Cambridge, The MIT Press, 2018, pp. 47-71.

[11] B. Jang, M. Kim and G. Harerimana, "Q-Learning Algorithms: A Comprehensive Classification and Applications," *IEEE Access,* vol. 7, pp. 133653-133667, 2019.

[12] Electricity Supply Board, "ESB Customer Supply Proposal for Smart Metering Tariff," ESB, South City, Ireland, 2010.