RESIDENTIAL DEMAND RESPONSE USING REINFORCEMENT LEARNING WITH PURSUIT ALGORITHM

PROJECT REPORT

submitted by

AMALA S RAJEEV

TKM21EEPS01

to

the APJ Abdul Kalam Technological University

in partial fulfillment of the requirements for the award of the

Degree

of

Master of Technology

in

Power Systems

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

T.K.M. COLLEGE OF ENGINEERING, KOLLAM

JULY, 2023

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING.

T.K.M COLLEGE OF ENGINEERING, KOLLAM

CERTIFICATE

This is to certify that the report entitled **'Residential Demand Response Using Reinforcement Learning With Pursuit Algorithm'** submitted by **Amala S Rajeev** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Electrical and Electronics Engineering is a bonafide record of the project work carried out by her under our guidance and supervision. This project report in any form has not been submitted to any other University or Institute for any purposes.

Dr. Imthias Ahamed
Project Guide
Professor
Centre for AI,TKMCE

[External Examiner]

Prof. Sumod Sundar	Prof. Jibi P Mathew	Dr. Sabeena Beevi K
Project Co-Guide	Project Coordinator	Head of Department
Asst. Professor	Asst. Professor	Associate Professor
Centre for AI,TKMCE	Dept of EEE, TKMCE	Dept of EEE, TKMCE

ACKNOWLEDGEMENT

First of all I am indebted to the **God Almighty** for giving me an opportunity to excel in my effort to complete this project on time.

I am extremely grateful to **Dr. T A Shahul Hameed**, Principal, TKM College of Engineering, and **Dr. Sabeena Beevi K**, Head of the Department, Department of Electrical and Electronics Engineering, for providing all required resources for successful completion of my project.

I am greatly obliged to my Project Guide, **Dr. Imthias Ahamed**, Professor, Centre for AI, TKMCE, for his encouragement and support.

My heartfelt gratitude to my Project Co-Guide, **Prof. Sumod Sundar**, Asst. Professor, Centre for AI, TKMCE, for his valuable suggestions and guidance in the preparation of the project report.

I express my sincere thanks to our project coordinator **Prof. Jibi P Mathew**, Asst. Professor of Department of Electrical and Electronics Engineering, and my friends for all help and co-ordination extended in bringing out this project successfully in time.

I will be failing in duty if I do not acknowledge with grateful thanks to the authors of the references and other literatures referred to in this project.

Last but not the least, I am very much thankful to my parents who guided me in every steps which I took.

Place: Kollam AMALA S RAJEEV

Date: July 2023 TKM21EEPS01

ABSTRACT

The primary goal of Demand Response (DR) is to lower the system's maximum demand. The introduction of smart grid and bidirectional communications make the implementation easier. A common way of cost minimization is shifting the loads from peak hours to off-peak hours. Reinforcement Learning (RL) is used for solving various optimization problem. Since the nature of the power system is stochastic, implementation of DR using RL techniques makes it more suitable. Here a scenario of scheduling residential loads with flexible devices is considered with the aim of minimization in energy consumption and minimum discomfort to the consumers. Q-learning which is a variant of RL is used for implementing the scheduling. The main concern is to find a balance between exploration and exploitation. One of the traditional RL methods is used for balancing exploration and exploitation is epsilon-greedy algorithm. The main challenge in the implementation of ϵ-greedy algorithm is to obtain the cooling schedule for balancing the exploration and exploitation. In this project, we propose an efficient algorithm for the action selection that is Pursuit algorithm. Here the performance of epsilon greedy is analyzed for various cooling schedule methods. The performance of RL algorithm using ϵ-greedy and pursuit algorithm is compared. The only parameter that depends on the performance of pursuit algorithm is the convergence rate β. As the dependency of pursuit algorithm on the hyperparameters is less and also no predefined episodes are required the convergence rate is faster than in ϵ-greedy. The performance of the algorithm is also analyzed using various tariff structures.

Contents

List of Figures

List of Tables

Abbreviations and Notations

1. DR - Demand Response
2. RL - Reinforcement Learning
3. HP - Hourly Price
4. GW - Grid World
5. DSM - Demand Side Management
6. RTP - Real Time Pricing
7. DLC - Direct Load Control
8. ANN - Artificial Neural Network

Chapter 1

Introduction

1.1 General Background

As we know that our traditional power system is just a one-way system. With the emergence of smart grid it enables the bidirectional flow of energy and informations. The major reasons for the introduction of smart grid are

- Environmental concerns
- Emergence of renewable resources
- Electronic communication technologies

The prime concern of the smart grid is to improve the efficiency of the overall system. Consumers also play significant roles to enhance the grid productivity. Before the introduction of smart grid, there are various ways to reduce the peak demand of the grid. However, those methods mainly focus on grid efficiency the consumer's comfort is not a concern. The emergence of renewable resources makes the existing methods more complex as the output of those sources are intermittent in nature. Widely used demand side management system (DSM) for the grid efficiency is demand response (DR) in a cost-effective manner. DSM along with Artifical Intelligence (AI) or Machine Learning (MI) tools makes it more easier and

convenient to implement compared to the traditional methods. The objective of the DSM programs is to reduce the peak demand of the grid. DR mainly focuses on the consumer's energy consumption style. There are various DR programs based on price and incentive. DR is employed in the residential as well as in industrial sectors. Since it's a financial benefit method for both the utility and the consumers, it encourages involvement [13].

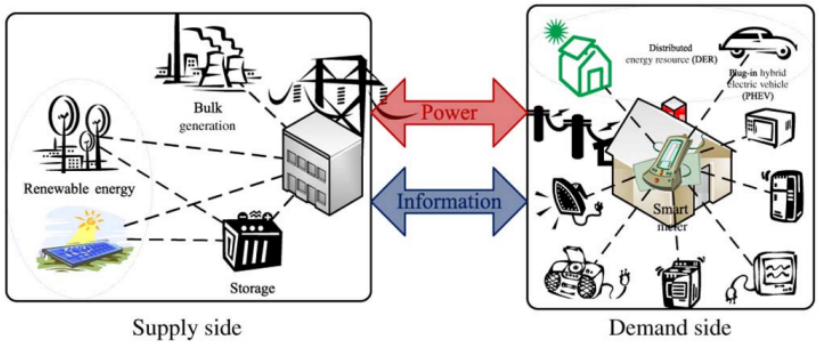

Fig. 1.1: Interaction between the demand and supply side

The commonly implemented DR is the price-based DR. There are various types of tariff structures, which makes it beneficial for residential consumers as well. The different types of tariff are Real time pricing, Time of Use, Flat rate tariff, Block rate tariff and Time of day tariff. The tariff structure is different for different consumers, it's based on their load consumption. For industrial consumers, the commonly used tariff is ToU tariff, for this kind of tariff the price of electricity is higher for peak periods and lower for off-peak hours. So based on the consumer's preferences and price, the load can be shifted from peak hours to off-peak hours. In real time pricing, the consumer and the operator exchange information about the price of electricity at that particular time. The consumers are informed from the utility side that they can participate in the DR event. In this thesis, we mainly focus on DR for the residential sector.

DR is easier to implement for small-scale industrial consumers and for residential sectors. For communication with the aggregator, home area network(HAN) is used. Various AI and MI algorithms, are used to implement DR. For the DR application in the residential sector, we apply Reinforcement Learning (RL) tech-

niques. Compared to other techniques, RL is suitable for obtain optimal solution in uncertain situations. In RL itself, there are various works, a few of them are explained in Chapter 2.

1.2 Motivation

The emergence of smart grids, along with the advancement in communication technology, made the implementation of DR programs possible. The main aim of DR is to reduce the maximum demand on the grid. Let us consider that there is a sudden increase in demand which is greater than the forecasted value. To meet the increased demand, the generation has to be increased. For meeting the demand by the conventional method is not a cost effective way so it can be achieved through various DSM programs. There are various controlling mechanisms to meet this requirement.However, with the complexities in power system infrastructure makes these methods much more difficult. The nature of power system is unpredictable so the mechanism used for controlling should be adaptable as well flexible. Electricity has become one of the basic needs of humans. Interruption of the supply discourages consumers from using the electrical energy from the utility. Therefore, they should also be aware of the critic condition. Hence, the introduction of programs like price or incentive based DR increases the participation of consumers. DR along with RL helps in obtaining an optimal solution. The random nature of the power system in consumption of consumers, tariff structures make RL more suitable for implementing DR. DR can formulated as Multistage decision making process.

1.3 Thesis Objectives

In RL there are various algorithms like Q-learning, SARSA, Monte Carlo, and Temporal difference method. Here we have implemented the load scheduling of residential devices using the Q-learning algorithm. One of the issues during the learning phase of the algorithm is action selection. It has been done using an exploratory policy. One of the main challenges of the exploratory is balancing between explo-

ration and exploitation. There are various methods like ϵ-greedy, Gibbs method and Pursuit algorithm for balancing it. Among these methods the commonly used one is the ϵ-greedy algorithm. Its well known that epsilon should be large at the beginning of the initial episode and decayed to a small value (0.001) at the end of the scheduling. But its not clear how to decrease epsilon, so people do experimentation which is known as cooling schedule and its a time consuming process. In this thesis we apply an efficient algorithm for the action selection i.e., Pursuit algorithm. Discomfort for the users vary from load to load. There are some works which consider how to balance the consumers discomfort and DR implementation.

The thesis objectives are given below:-

- To design a technique that reduces the total energy cost in a day with minimum discomfort to the consumers.
- To mitigate the problem of balancing between exploration and exploitation in ϵ greedy.
- To analyze the performance of pursuit algorithm for balancing exploration and exploitation.
- To analyze the effect of udc using both the algorithms.
- To compare epsilon greedy with pursuit algorithm

1.4 Structure of Thesis

The entire work is organized as follows.

Chapter 1: Introduction

This chapter deals with the various advancements in power system which leads to the implementation of DR possible. The reason for selecting RL among the machine learning tools for the application. This chapter also includes the motivation, objectives and structure of thesis.

Chapter 2: Literature Survey

This chapter deals with the various existing methods used for implementing the DR along with that a brief description of DR is also included. Since price based DR is the predominant one various tariff's are also discussed.

Chapter 3: Methodology

This chapter includes the methodology of the thesis. It gives a brief idea of RL with a simple Grid World (GW) example. It also includes the various RL parameters and how the design parameters are formulated for an RL problem.

Chapter 4: Implementation of the load scheduling using RL

This chapter deals with the main work of the thesis. The main challenge concerned with ϵ-greedy is overcomed using the pursuit algorithm. The effect of udc is also analyzed using the algorithms.

Chapter 5: Conclusion

This chapter deals with the inferences from the thesis and also the future scope and limitation of the work.

Chapter 2

Litearture Survey

This chapter deals with the various literature's reviewed for the implementation of Demand Response(DR) program. With the introduction of smart metering and bidirectional communication DR is becoming more popular. The basic need for DR is due to the integration of renewable source into the grid. As we know controlling these renewable sources are difficult as their output are intermittent and unpredictable. In this section, we also discuss about the DR and its various programs and tools that are commonly used for the execution.

2.1 Demand Response

Due to the increasing demand of electricity there are various emerging energy management tools such as Load shedding, Valley filling, Peak shaving to reduce the peak demand on the grid. These methods are the cost effective way of reducing the demand. The objective of DR is to balance generation and demand. Programs like demand side management (DSM) are introduced so that the maximum demand on the grid is reduced in a cost effective manner as well as it helps in increasing the participation of the consumers. Consumer participation are increased by providing incentives for those who participate in the DR event. DR encourages the consumer to change their consumption pattern according to the change in price of electricity.

Implementation of DR helps in increasing the efficiency and reliability of the grid.

2.1.1 Various DR Pograms

There are various DR programs available. The main classification is the Price-based DR programs and Incentive based DR programs. In Price-based DR programs the consumers shifts or reduce their load consumption's during the peak hours to off-peak hours. It is one of the traditional method for DR execution. Price-based DR program is again classified as Time of Use Tariff based, Critical Peak Pricing based and Real time Pricing [17]. This classification is based on the time differentiating pricing. In Incentive based DR programs the consumer inform their baseline details to the operators. In urge of any emergency situations, operators inform those consumers who can reduce their consumption during that period. If they are not able to reduce the demand then penalty will be charged else incentives will be provided. Incentive based DR programs are again classified into Direct-load control, Interruptible load services, Capacity market program and Demand bidding. Due to emergence of new technologies and user interface devices makes incentive-based DR more popular nowadays.

For the execution of the DR program in a residential sector, the consumer loads are classified as critical and non-critical loads, only the non-critical loads are shiftable during the DR event. While implementing DR in a residential sector the only prime concern is about the consumers comfort. In industrial sector the loads will be classified into shiftable equipments, non-shiftable equipments and controllable equipments. The only equipments that can be considered for the DR is the controllable and shiftable where the demand for non-shiftable should be continuously met. The primary concern of industrial demand response is the production target. Implementing DR in industries is complicated as compared to residential as it is mainly because of the industrial diversity and its production target. If any one of the machine is switched off for a longer period than the estimated time period it will result in a huge loss. Therefore various factors has to be considered while implementing DR in industries.

While considering the Price based DR programs, tariifs used should also be considered. Generally, tariff means the price of one unit electricity. There are various types of tariffs like Flat rate tariff, Two part tariff, Block tariff, Grid tariff, Variable-block tariff, Time-of-day tariff, Time-of-use(ToU) and Demand Conservation tariffs [13]. Among these tariff the commonly used tariff is two part tariff which consists of two parts fixed cost and variable cost. Time-of-day tariff is applicable for industrial consumers. In this tariff method the price is higher during the peak hours and lower during the off-peak hours. This method helps small scale industrial consumers to shift their load from peak to off peak hours. In ToU tariff different pricing structures are used for different seasons. In real time pricing the cost of energy varies with the actual production price of the utility.

2.2 Various DR Algorithms

There are various like Artificial Intelligence techniques and Machine Learning used for the implementation of DR. Here we discuss about the various tools for the implementation of DR in residential and industrial cases. The methods used for implementing DR is classified mainly into classical method and the heuristic method. Due to the integration of renewable sources makes the implementation of classical methods more complex. Heuristic approaches work more easier in these types of environments but the computational time gets increased with the number of variables. Different from the above methods RL is more suitable for such specified environments.

2.2.1 Industrial Consumers

There are several works that apply algorithm to solve DR in industrial methods. In all the methods they classify the loads into shiftable and non-shiftable loads in [10] and [4]. Shiftable loads are again classified into uncontrollable loads in [7]. In section, we discuss about various methods RL used for implementing DR.

In [10] proposed method classifies the consumer's load into two categories

mainly shiftable and non-shiftable. The classification is based on the function and the priority of the loads. High-priority loads are under the category of shiftable loads. Automatic switching off of loads during high-price periods. There is an agreement between the consumers and electrical utility based on the maximum demand acceptable range of changes are possible. An uninterruptible power supply is required by the non-shiftable equipment. For an industry the peak demand should always be less than the specified limits. Application of load shifting is more suitable than DLC. The tariff structure considered for the analysis is Multi-Year Tariff Order-2. EMS of the industry along with the smart meters helps in exchange of the information and synchronization in the operation.

Mainly there are two types of strategies:-shiftable class first strategy (SCF) and controllable device first strategy (CDF). The strategies used for load scheduling vary from load to load and the price of electricity. All the predefined strategies for each load are determined by the EMS operator. If the demand is greater than the reported demand to the utility, firstly the shiftable load with lower priority is switched off i.e., SCF strategy. Likewise, after switching off the load in each category it's checked whether the demand is within the permissible limit or not. Even though the demand is not falling the next option is lowering the operating range of controllable equipment i.e., CDF strategy.. The process terminates when the demand is within the acceptable range. Here the algorithm requires some predefined parameters like the consumption details, consumption details, and load category details. Along with this, the main information required is the maximum demand that's agreed to the utility. In order to decide the shifting of the loads a shifting factor is considered for both the loads. Shifting is done using XAMPP software. Based on the requirements of the consumers any one of the strategies can be employed at a time. The embedded software along with the algorithm analyzes the process for every half hour and samples the information for further analysis also. Implementation of the algorithm saved around 25% of energy costs.

In [4] DR is implemented using the State-Task Network (STN). They have also considered the production requirements to be satisfied. The loads are classified

based on the priority of operation, task requirements to be satisfied and production targets. Through the wide area network the energy management system communicates with the utility and the day ahead schedule of the loads is obtained. The network mainly consists of two nodes- one is representing the input, intermediate, and output and the other node is used for representing the operations. The framework consists of two models- the utility and the demand side. The industrial communicates the utility through the Wide Area Network (WAN). The electrical equipments are classified into non-shiftable and shiftable. The production industry is classified into two parts- The production unit and the EMS. EMS schedules the equipment based on the day-ahead energy price obtained from the utility.
For the DR implementation, the task of the industry is divided into schedulable and non-schedulable tasks. While formulating the functions the constraints considered are operating constraints for schedulable tasks and the material balance in each stage. For each operating point, there are two sets of parameters. The main set includes the production rate, consumption of raw materials, and electrical energy demand for that task. The second parameter set includes the time interval of operation and the shutdown. It should be ensured that the required amount of materials are obtained to the next stage of operation. Maximum demand should be less than the specified limit to the utility for that purpose a maximum demand limit indicator is also used. Implementation of the DR using day-ahead scheduling in the flour industry helped the consumer to attain 38% savings.

In [7] the equipments are classified into shiftable, nonshiftable and controllable. The energy demand for the non-shiftable equipment should always be met continuously irrespective of the electricity price. Only the scheduling of shiftable and controllable equipments are available for scheduling. In the case of schiftable equipments their entire operation can be shifted whereas for controllable equipment their operating points can be varied. The EMS of the industry communicates with the utility through WAN and the communication to various departments of the industry is through Local Area Network (LAN). Here along with the primary objective they also consider the production requirements. Here the analysis is done

on an hourly basis. A day is discretized into 24 intervals and the scheduling for the respective equipments is obtained. Production-Resource is balanced in such a way that the quantity of a resource stored at the terminating stage depends only on the amount stored during the previous stage and new production and utilization during the present stage.

Along with the primary objective of cost minimization, they also consider that the production target is achieved. At the end of the process, the final residual storage must be greater than or equal to the target set. Deep RL approach is used for the implementation. State formulation consists of a combination of external and internal parameter. The external parameter includes the hourly price of electricity and the internal parameter includes the residues obtained from each stage. Action set includes the operating points of the controllable and the shiftable loads. Reward function is formulated in such a way that it considers both the energy cost and production of resources. When the state space becomes larger its difficult to use the Q-learning method or SARSA, the complexity increases and the time required for convergence also increases. So to avoid those difficulties the problem is solved using Actor-Critic based Deep RL method, where the actor takes all the actions and the critic uses the value function to evaluate the policy. When the dimensionality of state space increases approximation using the linear function is difficult so to approximate the policy and the value function neural networks are used.

2.2.2 Residential Consumers

There are several works which apply various algorithm to solve DR in residential methods. There are several methods uses RL [5],[1],[9],[6] and [11] and non-cooperative game theory in [2]. Most of them not considered the discomfort in [2] but some of them have considered the discomfort [5],[1],[9],[6] and [11]. Brief introduction is given below.

In [11] Residential Consumers solely depending on the utility is considered. Here the authors propose a batch RL method for the DR implementation of residential thermostatically controllable loads. Convergence rate of batch RL is faster

compared to Q-learning and SARSA its because of the ability of the algorithm to use the past datas. The authors have considered a water heater and heat pump. The scheduling of these equipment is obtained a day prior using the batch RL method. For the safety of the users, an inbuilt control is also provided which will overrule the agent in unsatisfactory conditions. Feature extraction is used for extracting the hidden variable or to reduce the state vector dimensions. Here there are two models been considered one is dynamic pricing and the other is day-ahead scheduling. For dynamic pricing, the policy is based on the present value of the state i.e., it's a closed-loop policy. The actions are based on the external signal which is the price of electricity. Whereas for the day ahead scheduling its open-loop policy, it does not depend on the upcoming value of the state. Along with the main objective of cost minimization they also aim to reduce the deviation of the day-ahead schedule obtained using the method and the actual consumption of the users. Day-ahead schedule is obtained using the batch RL algorithm and model free-Monte Carlo method.

In [6] the authors propose a Q-learning algorithm along with artificial neural networks (ANN) for the scheduling of controllable loads in the premises of the consumers. As we know the main aim of the Home Energy Management System (HEMS) is to obtain optimal load scheduling by continuous monitoring of the load details from the smart meter. To this existing system, they integrated the RL technique to achieve the objective and also by considering the consumer's comfort. Here they consider a smart home with rooftop solar and all the appliance operations are independent of each other. Operating points of devices like air conditioning devices are obtained from the ANN. It also helps to improve the performance of the appliances.

The appliances are classified into controllable and uncontrollable devices. Under the category of controllable devices it includes the shiftable appliances and reducible appliances. Reducible appliances mean thermostatic loads whose consumption can be reduced to minimize the cost. Shiftable appliances are categorized into interruptible and uninterruptible. For uninterruptible appliances, their operations are

not interrupted by the HEMS. Uncontrollable appliances are not considered for the scheduling. The conventional way of obtaining the indoor temperature is by approximation of thermal parameters. State space includes the states of the washing machine, AC, and state of charge of the energy storage devices. The action space available for AC is turn or off Whereas for AC it's been divided into 10. Action space for ESS consists of charging, discharging, and idle. Reward function formulation for each load consists of their energy cost along by considering the undesirable operation of the washing machine, thermal comfort of AC, and overcharging or undercharging condition of ESS. The penalty is assigned for unsatisfactory operations. The transfer function used for the operation of ANN is RELU. Optimisation is done with Adam optimization algorithm. The performance of the algorithm is observed to be more efficient than the existing conventional approach.

In [8] the authors have considered each participant of the program as a smart home energy management system (SHEMS). Due to the increase in the integration of renewable sources, each user is assumed to be with backup storage like batteries. When the real time price of electricity goes high then the consumers can switch to backup storage. When price is low then that period can be utilise for charging the battery. Therefore the authors have proposed a modified distributed algorithm that is based on dual decomposition. In this method the problem is solved by subdividing it into various independent problem. Hence it helps in determining the energy demand, charging and discharging schedules and the energy supply. The algorithm also ensures that the privacy of the consumers is secured which promotes their participation.

Smart meters installed at the consumer's premises is also provided with an energy consumption controller (ECC). This specific unit plays the role of obtaining the optimal scheduling of the units. Along with the primary objective of cost reduction this method also aims to increase the social welfare between the utility and the consumers. The analysis is done on an hourly basis. A day is divided into 24 hours with the help of ECC the optimal scheduling for that particular time period is obtained. The utility will be informing the price before the beginning of that

particular slot and it remains unchanged during that slot. For the execution of the distributed algorithm, initially, all the users upload their demand, energy for charging and discharging the battery, and the calculated parameters to the utility company through the smart meters. Based on all the information obtained from the users the utility calculates the price of energy and forwards it to the consumers. Since the nature of the system is stochastic it is solved using receding horizon optimization.

In [5] along with the cost minimization they have also considered the thermal comfort level of the users. The non-linear characteristics of the HVAC system made them more complicated to be controlled by PID or any other logic. So the authors proposed the implementation using reinforcement learning. They have considered mainly two scenarios that are scheduling the loads with and without DR along with RL. DR can be implemented using any one of the two ways in a HVAC building i.e., Setpoint method or DLC. However, using DLC is faster but while considering the comfort level of the consumers it's not a better option. Load shifting method can also be implemented by identifying the zone which requires more energy. Pre-heating or cooling those zones based on the application during the off peak periods. During the peak hours it requires only lesser energy. here they focus DR using the set point as an effective way.

They considered mainly two frameworks- Modelling the building and its controlling unit then the RL framework. For the modeling of building, there are various stages like creating an energy-plus building, finding an appropriate RL agent for the problem then running the simulation for various scenarios. In order to attain maximum utilization of energy a penalty factor is introduced i.e., HVAC penalty factor. Since the comfort of the consumers is also considered, thermal penalty factor is also included in the reward formulation. For the analysis purpose they have considered a five-zone air-cooled building. Various modes of operation are considered. Mode-1:- HVAC is off and the comfort of the consumers is given priority. Mode-2:- HVAC power is in use and consumers priority is given more importance. Mode-3:- Since the HVAC is ON, now there are two priorities i.e., the energy efficiency and the

priority of the consumers. Mode-4:- If the HVAC power consumption is reduced by keeping the consumers comfort so the controller tries to minimize the adverse impact on the thermal comfort of consumers. Mode-5:- When the DR is activated and HVAC is ON then priority is given for the cost reduction. Using this method saved more energy than just by using the energy plus controller.

In [2] the main application of non-cooperative games is in obtaining the optimal solution for residential energy management. Along with the main objective, the comfort of the consumers is also considered. Here the basic building blocks are the load model and the utility model along with the total cost function. For the formation of the load model, the loads are classified into shiftable and interruptible loads. There is predefined scheduling for the shiftable equipment. Hence the load models for those loads are represented as binary variables. It represents the period for which the equipment is in operation. In the case of thermostatic load, there is a permissible limit in the temperature conditions which are based on the owner's preferences. Scheduling of the uncontrollable loads is considered as deterministic it's done based on past consumption. Integration of the distributed energy resources are also possible with the proposed model.
Cost function takes into consideration of the electricity bill of consumers and the willingness to take part in DR. For the formulation of the cost function two models are considered one is a quadratic and peak-pricing. Since the fixed component is not affected by the scheduling of loads only the variable part of quadratic cost function is considered. The common residential tariff is the peak pricing where the cost includes the combined prices of energy and demand cost. The methods provides the participants of the events to interact and share information. The billing scheme for the cost formulation is obtained either by proportional to consumption (PTC) or peak-time-slot (PTS). The key idea behind PTC is that the total cost paid by all the consumers is equal to the total cost incurred by the utility. In PTS the pricing is based on the consumption for each slot. If the load model, total cost function model, and billing mechanism are obtained then using the non-cooperative gaming theory optimized cost can be obtained. The process proceeds iteratively and terminates

when the required equilibrium is obtained.

In [1] HEMS provides users to participate in DR events. The aim is the shifting of the loads from peak hours to off-peak hours. For the analysis the authors classified the residential loads into shiftable and n0n-shiftable appliances. Operation of the schedulable appliances is based on the RTP received by the smart meter installed at the user's premises. Here they implemented the scheduling using the traditional Q-learning method. The state space includes the energy price and the demand. Demand is classified into 3 levels i.e., high, medium, low. Energy price is classified into expensive and cheap. Action space consists of mainly three actions i.e., do nothing, shifting, valley filling. Valley filling means the shifting of the high-priority load during the off-peak hours to reduce the demand of the consumers. Fuzzy logic interface evaluates the actions. The reward function formulation is done with Fuzzy reasoning. Here for the analysis they used Mamdani-type FIS. The input to the FIS is the load specifications and the electricity pricing details. The output is anyone of the actions specified in the action space.
Suppose if an action is taken in a particular state then the output of the FIS will be very good action, good action, or bad action. The membership value of the fuzzy is in a range of [0 100] so that all the possible actions are explored. Here they have also considered that the consumer's requirements should be satisfied while scheduling the loads. This RL technique along with fuzzy reasoning helps to smooth the power consumption profile along with cost reduction.

In [9], the authors introduced a modified incentive-based behavioral RL approach. Along with this behavioral approach residential DR using RL with pursuit algorithm for action selection, they considered how the incentives affect the consumer's willingness to participate in the event. For accurate analysis, they integrate the proposed method with the Monte-Carlo method. The components of the system considered for the reliability analysis are Generating units and Advanced Metering Infrastructure (AMI). From the DR point of view, the modeling done are load characteristics and willingness of consumers. For the modeling of generation units, the factors considered are mechanical availability and the supply of fuel. MDP is used

for the modeling of the mechanical framework. AMI plays an important role in enhancing the reliability of the system. It provides bidirectional communication and gathers demand data and other operational availabilities. AMI is modeled using a two-state model.

Loads are categorized into critical, shiftable, and interruptible. Demand for the critical has to be continuously met. Interruptible loads can be partially or fully interrupted. For the modeling of interruptible loads, a willigness factor(WF) is also introduced. It indicates the willingness of the consumers to participate in the DR. The value varies from consumer to consumers therefore its been treated as a stochastic variable. The method used for the modeling of WF is regret matching procedure(RMP). It is an economic analysis that helps in determining the behavior of the consumers easily. Consumers are provided with a pay-off to increase their participation in DR. After obtaining the WF for each consumer for a particular interval, DR event possibility is obtained from the AMI. If there is a chance of shortfall then the loads are scheduled on 24hr basis. From the modified profile of DR, the reliability is analysed. Various factors have been considered for the analysis are energy not supplied, loss of load duration, and loss of load occurrence. After each DR event, those indices are compared with previous values.

2.3 Conclusion

In this chapter we have seen various tools used for implementing the DR. Commonly used DR program is Price based DR. Introduction of various tariff structures makes it more beneficial to the residential consumers. The primary concern of all the authors is to minimize the consumption. Only few of them considers about the safety of the consumers and their comfort level. Execution of any methods without considering the consumers satisfaction is not an ideal solution. It may also cause the consumers to not participate in DR events. Therefore the consumer satisfaction should be considered along with the primary objective.

Chapter 3

Methodology: Reinforcement Learning

3.1 Introduction

Reinforcement learning is a type of machine learning used to solve many optimization problems [15]. The major problem concerning the power system is maintaining the balance between demand and generation. As we know, both are stochastic in nature, choosing RL is a better option. Finding the optimal solution under uncertain conditions makes RL more suitable compared to other optimization techniques. Another advantage of RL is its adaptability to any environment. In this section, let's have a brief idea about RL, its related terms, and how it can be useful in solving various power system problems

3.2 Grid world using RL

There are various ways of solving optimization problems. When compared all other optimization techniques, RL is more adaptable to stochastic conditions. Basic elements of RL are an agent and environment. Agent is the one that makes the decision and takes the actions where environment is all the other things outside the

agent. It's one of the potential tool in solving uncertain decision making problems. Here the agent is the exploratory policy that is been used to take the optimal actions and Grid cells indicates the states of the environment. As a intro to RL lets consider the case of a grid world problem. The GW that is been considered for this specific scenario is shown in Fig.4.1. Here the grid world problem can be viewed as a Multistage decision making problem. Its a 6*6 grid world. The objective is to find sequence of actions to reach the goal state in shortest path and with minimum expected cost.

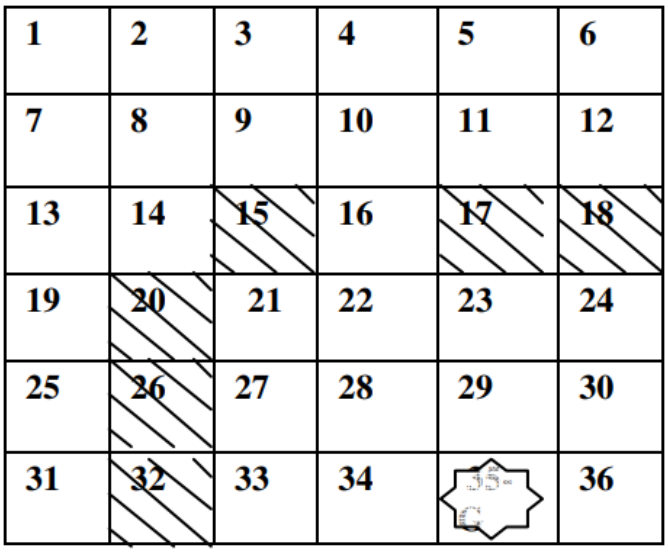

Fig. 3.1: Grid World

3.2.1 RL for solving MDP: Design Parameters

STATE

State consists of information about decisions to be taken [18]. In this GW problem the number of states are 36 (1-36). Therefore the state space X={1,2,3...,36}. The goal state or the terminating state is 35 (x_g). Here the states are represented using one dimensional vector it can also be multidimensional like the representation of a state using its specific row and column.

ACTION

It indicates the actions that are possible in all the states. Here we are there

are four possible actions in every state i.e., left, right, up, and down. For the analysis the action set A has been formulated as A={1,2,3,4} where "1" represents right action, "2" represents left action and so on. Diagonal actions are also possible which results in 8 actions in-total for a particular state. For the easiness of the explanation and analysis, we are considering only four states.

TRANSITION FUNCTION

If an action is taken in a particular state the new state is obtained using the transition function. Lets consider the transition function

- $(x_k,a_{right})=x_k+1$
- $(x_k,a_{left})=x_k$-1
- $(x_k,a_{up})=x_k$-6
- $(x_k,a_{down})=x_k+6$

However there are exceptional cases for the upper left and right corner state {1,6}, lower left and right corner state {31,36}, upper and lower boundaries {2,3,4,5, 32, 33, 34,35} then the left and right boundaries {7,13,19,25,12,18,24,30}. Suppose if the action taken in state "1" is left then the agent should remain in state "1" itself likewise an action of up in upper boundaries states will result to remain in that particular state itself. In the case of left boundary states an action left will result to remain in that state itself and in an action of right in right boundaries causes the agent to remain in that state itself.

REWARD FUNCTION

Reward mainly describes about the goal of RL problem. It helps the agent to differentiate between the good and bad events. From Fig.3.1 we can see that there are two types of blocks one is shaded and another is unshaded. Our objective is to find the shortest path with minimum expected cost. Cost function is formulated in a way that the cost of shaded blocks are set to high (here its 10) and that of

unshaded blocks are set to low (here its 1).

$$g(x, a, x_{new}) = \begin{cases} 1, & if x_{new} \text{ is in one of the unshaded blocks} \\ 10, & if x_{new} \text{ is in one of the shaded blocks} \end{cases} \tag{3.1}$$

Q-LEARNING

Q value for a state action pair (x, a) means the total expected cost when we start from state x, take an action, and then follow the optimal policy, Π^* [15].

$$Q(x, a) = E\{\sum_{k=0}^{N-1} g(x_k, a_k, x_{k+1})\} \tag{3.2}$$

Basically policy is a mapping from states of the environment to actions to be taken in those states. Here there are four possible actions for a particular state therefore corresponding to one state there will be four Q values. Q values for a particular state x' is given as Q(x',a_1),Q(x',a_2),Q(x',a_3),Q(x',a_4). The Q values quantitatively indicates how good is to take that action in that particular state. All these Q values are stored in a look-up table. In state x',a^* is said to be the optimal action if Q(x',a^*)<Q(x',a_i) for all $a_i \neq a^*$, then the optimal action is said to be a^*, where its defined as

$$a^* = \arg\min_{a'} Q(x', a') \tag{3.3}$$

Inorder to learn the optimal actions in each states the Q values should be known in advance. Therefore learning the Q values is a challenging task its done with exploratory policy. To learn the Q-values we begin by initialising them to random values or zeroes. Once an action is taken in a particular state then the Q values are updated using the Bellman equation.

$$Q^{n+1}(x_k, a_k) = Q^n(x_k, a_k) + \alpha[g(x, a, x_{new}) + \gamma \min_{a'} Q(x_{k+1}, a_k) - Q^n(x_k, a_k)] \tag{3.4}$$

Here "α" indicates the learning rate which implies that how much the Q values can be modified. The Q values are updated till the agent reaches the goal state. So it's clear that the action selection is depend on new parameter α. If the value is decayed gradually the Q values converge to optimal values. But when the value of α becomes very small the adaptability of the algorithm decreases. As we know that

many of the power system problems are stochastic therefore the algorithm should be adaptable to any environment. If we keep the value of α to a fixed value there are chances for the algorithm to diverge but it may be adaptable. This is also not a satisfactory condition.

Algorithm 1: **Implementation of RL using Q-Learning Algorithm**

Initialize Q(x,a) to zero for all (x,a) pairs
Initialize the maxepisodes, learning rate and penalty.
for 1 to maximum episodes **do**
    step=0
    x=starting state
    while (x!=x_g) **do**
        Select an action "a" based on exploratory policy
        Obtain the new state using the transition function
        Calculate the cost using g(x,a,x_{new})
        Updating the Q values using Bellman Equation
        x=x_{new}
        k=k+1
    end while
    Total step = k
end for

It concludes that the selection of action is the prime concern in RL that is there must be balance between exploration and exploitation. If the Q values are prior known then its not an issue. Initially the Q values are unknown, therefore the actions that are taken to be the best may not be the best action. The traditional way of balancing this issue of exploration and exploitation is to use the Epsilon greedy algorithm. Other methods are the Gibbs method and the Pursuit algorithm.
In epsilon-greedy algorithm initially the value of ϵ is chosen to high and later its decayed. Initially, when the value of ϵ is very high then the probability of taking the random action is ϵ and that of the greedy action is 1-ϵ. A detailed explanation

of these algorithms is given in Chapter 5.

3.3 Conclusion

In this chapter, we have seen a brief introduction to RL. Like a grid world problem, our real time power system problem can be modeled as a MDP. If a problem can be formulated as MDP then finding the optimal solution with RL is easier. Various Q-learning algorithms, are explained in detail in the upcoming chapters.

Chapter 4

Implementation of load scheduling using RL algorithm

4.1 Introduction

This chapter deals with scheduling of the residential load using epsilon greedy and pursuit algorithm. The performance of these algorithms depends on various parameters. The decay method of epsilon in epsilon greedy algorithm is of prime importance. There are various types of decay methods like exponential decay method, discrete decay methods and linear decay methods. Among these three methods the commonly used decay method is the exponential decay method, based on the application the usage of these methods varies. Here the analysis of epsilon greedy using exponential decay and discrete decay method has been done. In the pursuit algorithm the convergence rate β is the important parameter. The performance of pursuit algorithm for various values of β has been analysed.

4.2 Problem Formulation

For this particular case we are considering "m" number of flexible loads based on the assumption as given below [16].

- All the loads are independent of each other.
- Fixed Power consumption 'rt'
- Start time 's' and finish time 'f', total duration is 'l'
- Discomfort due to delay in switching is considered using 'udc'

When the value of udc is zero or very low then there is minimum discomfort for scheduling the loads. The loads to be scheduled can be modelled as L_j=(s_j,f_j,l_j, rt_j,udc_j). Here the main objective is to minimize the cost of electricity in a day with minimum discomfort of the consumer due to scheduling. Since we have to minimize the cost, let the cost of one unit of electricity during the k interval be C^k. Here the decision to be taken is whether to switch the load on or off. It can be represented using a decision variable u_j^k. u_j^k=1 indicates that the load j is switched on during the period k. The function that has to be minimized can be formulated as

$$\sum_{j=1}^{m}\sum_{k=1}^{24} C^k rt_j u_j^k \tag{4.1}$$

Subjected to the constraints

$$\sum_{k=1}^{24} u_j^k = l_j \tag{4.2}$$

u_j^k=0 $if\ k < s_j\ or\ k > f_j$

In our assumption we have considered that all the loads are independent of each other i.e., the scheduling of one load does not affect the other, and also solving two minimization problems is a complicated task, the objective function can be solved as "m" independent equations. Therefore the objective for a specific load can be rewritten to

$$\sum_{k=1}^{24} C^k rt_j u_j^k \tag{4.3}$$

Subjected constraints remains the same as in eqn.(4.2)

For the analysis purpose, consider the case of five residential flexible loads whose operations are independent of each other. The specifications of the loads are

given in Table 4.1. Price of electricity considered for the analysis is given in Table 4.2

Table 4.1: Load specifications

Load	**s**	**f**	**l**	**rt**	**udc**
D_1	8	15	4	10	20
D_2	8	15	4	10	0
D_3	11	19	5	7	2
D_4	10	24	7	5	1
D_5	6	18	8	5	3

Table 4.2: Price of Electricity

Time	**Price of Electricity**
1-10	5
11-13	12
14-19	5
20-22	10
23,24	5

4.3 RL algorithm based on Epsilon-Greedy

Solving an optimization problem using RL requires the formulation of State, Action, Reward, State transition Function, Reward function, and Action selection strategy. Here we discuss how the design parameters of RL are chosen based on this specific application i.e., load scheduling.

4.3.1 Design Parameters

STATE:- The decision to be taken depends on the information contained in the state. As the main objective is to reduce the total energy cost the decision has to be taken such that the objective is achieved. Therefore the state mainly depends on the electricity price which in-turn depends on the present time of the day and

the number of the hours the load is switched on. Hence the state consists of two variables x_1 and x_2 where x_1 denotes the current time of the day and x_2 denotes the number of hours the load is turned on. So the state space can be represented as X={(x_1,x_2) s.t x_1=s_j,.....,f_j ; x_2=0,.....,l_j }

ACTION:- Mainly there are two possible actions in every state. Either the load can be switched on or switched off based on the functions and the constraints. The action set A consists of A={0,1}.

STATE TRANSITION FUNCTION If we consider a particular load the scheduling begins from the starting time "s". The scheduling begins at state (s_j,0). As the variable x_1 is independent of the action and depends only on the time. Hence it moves forward $x_{k+1}(1)$=$x_k(1)$+1. The variable x_2 depends on the action that's been taken. Therefore generalizing the transition of x_2 as $x_{k+1}(2)$=$x_k(2)$+a.

REWARD FUNCTION:- While moving from one state to another the cost incurred has been formulated using the function g(x,a,x_{new}). Where,

$$g(x,a,x_{new}) = \begin{cases} C^k rt_j, & if a = 1 \\ udc_j rt_j, & if a = 0 \\ penalty, & if x_{new}(1)\text{=f AND } x_{new}(2)\text{<l} \end{cases}$$

If udc=0 then $\sum g(x,a,x_{new})$ is the total energy consumption cost and if the constraints are violated then then cost increases

If udc is not equal to zero then the reward will be equal to is the sum of price of electricity and kl*udc, where kl is the number of hours when load remained off.

Penalty is given to avoid the execution of unwanted actions.

ACTION SELECTION STRATEGY:- In RL selection of actions during the learning stage of the algorithm is an important issue. Q values are defined for a state-action pair. For every state there will be an optimal action a^*. If the agent is in state x, it can take two actions i.e., a_{on} or a_{off}. Consider that the Q values are aprior known then the optimal action is said to be a_{on} if $Q(x, a_{off}) > Q(x, a_{on})$.

Therefore the optimal action a^* for any state can be represented in general as

$$a^* = \arg\min_{a'} Q(x', a') \tag{4.4}$$

. However the Q values are not known in prior action selection is based on the exploratory policy. Balancing between exploration and exploitation is done using ϵ-greedy algorithm.

4.3.2 Epsilon-Greedy Algorithm

The main objective of epsilon greedy algorithm is to balance between exploration and exploitation. Epsilon greedy algorithm tries to select the greedy action based on the estimated reward. Various parameters which affects the performance of the algorithm are learning rate (α), epsilon (ϵ) and discount rate (γ). Among these parameters ϵ is of more importance. Initially the value of ϵ is chosen high nearly 0.9 (where the value of ϵ can vary from 0 to 1). Probability of choosing a random action is ϵ say 0.9 and that of greedy action is 1-ϵ say 0.1. The reason for choosing the value of ϵ close to 1 initially is to explore the defined environment. Later the value of epsilon is decayed to increase the exploitation. Decaying rate of ϵ is obtained by trial and error method known as cooling schedule. There are various decay methods like the exponential decay method, discrete decay method and the linear decay method.

4.3.3 Implementation of RL using Epsilon Greedy Algorithm

Algorithm 2: **Implementation of RL using ϵ-greedy**

Define the environment
Initialise the maxepisodes, learning rate and penalty.
for 1 to number of flexible loads **do**
  Initialise the Q values to zero
  Initialise the value of epsilon
  for i=1 to maxepisodes **do**
    Initialise the state vector
    while (x(1)<f and x(2)<l) **do**
      Selecting greedy action with a probability of 1-ϵ or
      random action with a probability of ϵ
      Obtain the new state using the transition function
      Calculate the cost using g(x,a,x_{new})
      Q(x,a)=Q(x,a)+α[r+γ max Q(x_{new},a)-Q(x,a)]
    end while
    ϵ=ϵ*decay rate
  end for
  Storing the schedule for that particular load
end for

4.3.4 Results of Epsilon Greedy Algorithm

Implemented the load scheduling using ϵ-greedy algorithm. The parameters chosen for the implementation are given in Table.4.3.

Table 4.3: Parameter considered for ϵ-greedy algorithm

Parameters	Values
α	0.1
γ	1
ϵ	0.99
Episodes	5000

TIME	D1	D2	D3	D4	D5
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	1
8	0	0	0	0	1
9	1	0	0	0	1
10	1	1	0	0	1
11	1	0	0	0	0
12	1	0	0	0	0
13	0	0	0	0	0
14	0	1	1	1	1
15	0	1	1	1	1
16	0	1	1	1	1
17	0	0	1	1	1
18	0	0	1	1	0
19	0	0	0	1	0
20	0	0	0	0	0
21	0	0	0	0	0
22	0	0	0	0	0
23	0	0	0	1	0
24	0	0	0	0	0

Fig. 4.1: Scheduling of all the loads

For Load-1 from the table 4.1 it is clear that the udc factor is very high. Therefore the load scheduling has been done for the consecutive 4 hour duration and the price of electricity obtained is also high. Therefore it indicates that when the "udc" is high then the scheduling is done irrespective of the price of energy. The action sequence for that particular load is given in table 4.3. Plot of variation ϵ vs episode is given in Fig.4.2

The total cost of energy obtained after implementation of the algorithm is 1170/-. If the loads are operated without DR then the cost obtained is around 1507/-. Therefore there is a total savings of 22.36%. Action sequence obtained after scheduling the load is given in Table 4.4

Table 4.4: Action sequence of each load

Load	**(s,f,l,rt,udc)**	**Action Sequence**	**Cost**
1	(8,15,4,10,20)	8,9,10,11	300
2	(8,15,4,10,0)	8,9,14,15	200
3	(11,19,5,7,2)	14,15,16,17,18	175
4	(10,24,7,5,1)	14-19,23	175
5	(6,18,8,5,3)	6,7,8,9,14-17	320

From table 4.1 its clear that the specifications of load-1 and load-2 are the same, only difference is in udc. Since the udc of load-1 is high, the scheduling is done during the period 8,9,10 and 11. For load-2 the scheduling is done during 8,9,14 and 15 which is the minimum energy cost period.

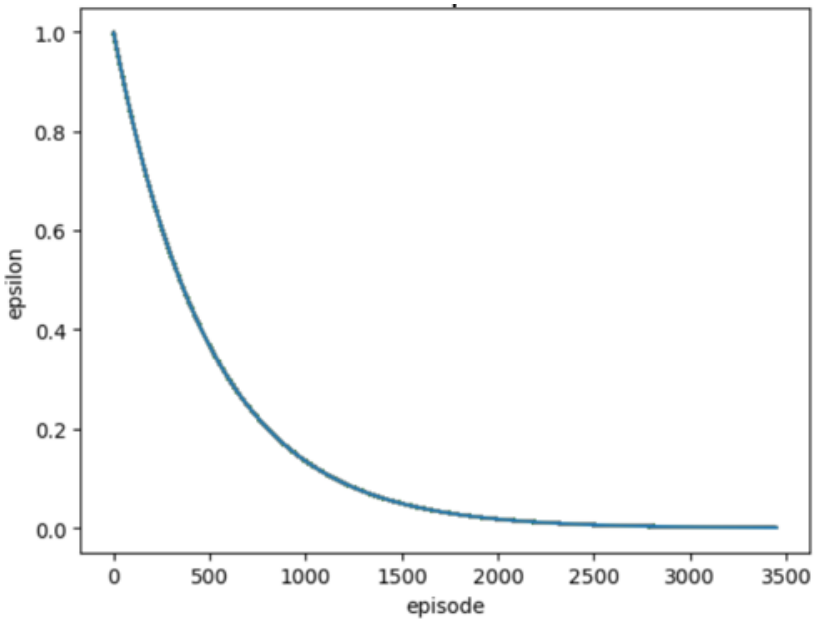

Fig. 4.2: Exponential decay of ϵ after each episode

Analysing the effect of udc

Table 4.5: Load Scheduling when udc=0 for all loads

Load	**(s,f,l,rt)**	**Load ON**
1	(8,15,4,10)	8,9,14,15
2	(8,15,4,10)	8,9,14,15
3	(11,19,5,7)	14,15,16,17,18
4	(10,24,7,5)	14,15,16,17,18,19,23
5	(6,18,8,5)	6,7,8,9,14,15,16,17

When the value of udc=0 for all the loads are scheduled during the period when the cost of energy is very low. For load-1 the load has to be scheduled for a 4 hour duration starting from 8-15. Since the value of udc=0, the load is on for the periods 8,9,14, and 15.

Table 4.6: Load Scheduling when udc=7 for all loads

Load	**(s,f,l,rt)**	**Load ON**
1	(8,15,4,10)	8,9,10,11
2	(8,15,4,10)	8,9,10,11
3	(11,19,5,7)	11,12,13,14,15
4	(10,24,7,5)	10,11,12,13,14,15,16
5	(6,18,8,5)	6,7,8,9,10,11,12,13

When the value of udc=7 for all the loads are scheduled in operation for consecutive hours. For load-1 the load has to be scheduled for a 4 hour duration starting from 8-15. Since the value of udc=7, the load is on for the periods 8,9,10, and 11. Here the priority is for the consumers comfort.

Table 4.7: Effect of udc

udc	**Total energy cost**
0	945
3	1282
5	1460
7	1538
9	1538

The value of 'udc' is varied for all the loads. For analyzing the effect of udc the udc is varied from 0 to 9. When the value of udc is kept very low for all the loads, the scheduling is done such that the cost is minimised. When the value of udc is kept very high for all the loads the scheduling is done for the respective consecutive hours. So when udc is higher then udc is given more priority. This particular analysis also helps in identifying the range of udc. From the table 5.6 its clearly evident that when then udc value is increased above a certain limit its effect is constant on the total cost.

Comparison between various decay methods in epsilon-greedy

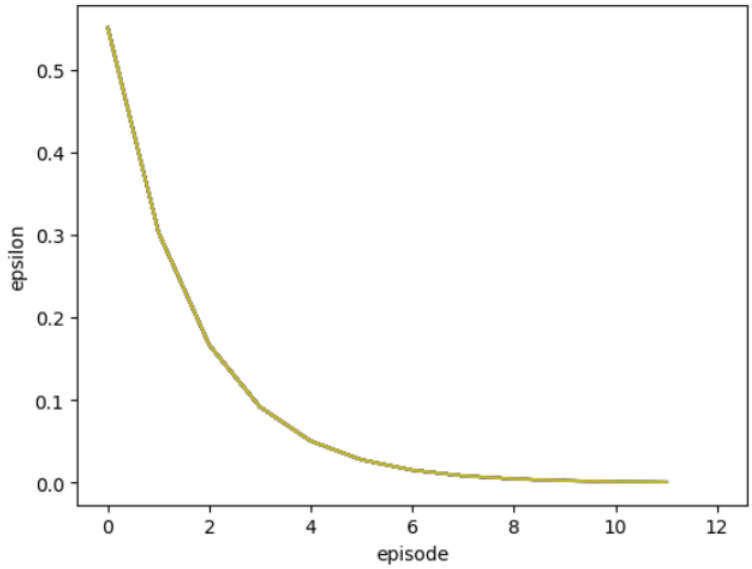

Fig. 4.3: Cooling schedule-1 of exponential decay method with decay rate=0.55

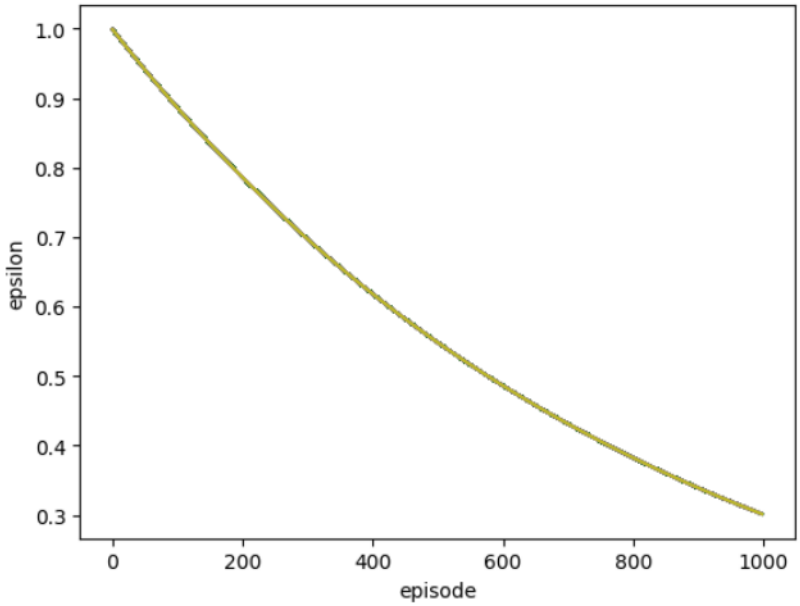

Fig. 4.4: Cooling schedule-2 of exponential decay method with decay rate=0.85

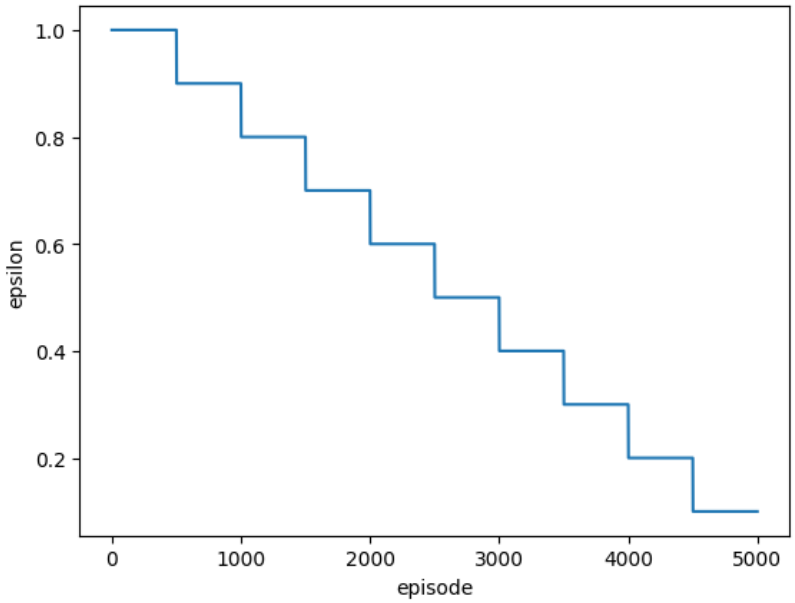

Fig. 4.5: Cooling schedule-3 of discrete decay when no. of steps=10

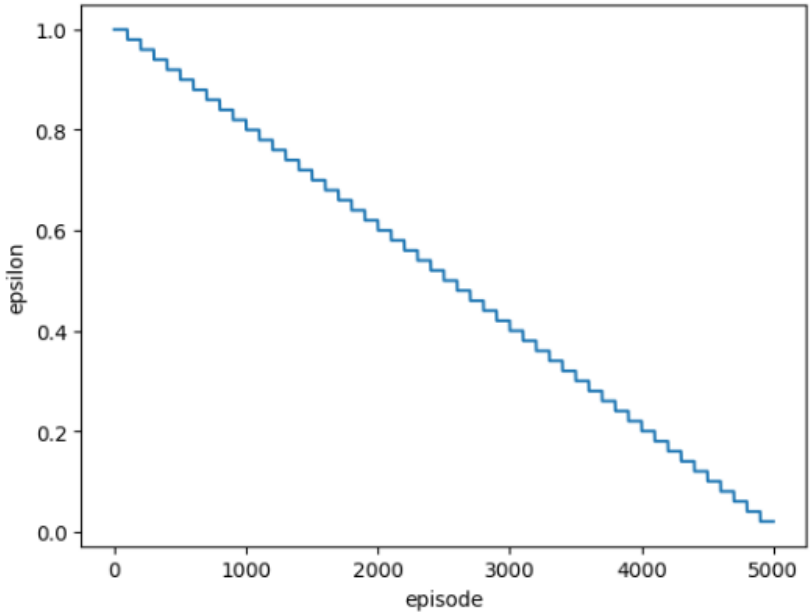

Fig. 4.6: Cooling schedule-4 of discrete decay when no. of steps=50

Table 4.8: Results of Discrete decay method

No. of steps	No. of Episodes	Cost
10	5500	1190
20	5000	1185
50	5000	1170
70	5000	1170

Table 4.9: Comparison of Decay methods

Exponential Decay	No. of Episodes	Cost
0.5 (α=0.01)	3500	1500
0.85 (α=0.01)	3500	1235
0.5(α=0.1)	1000	1150
0.99(α=0.1)	1000	1170

In cooling schedule-1 & 2 the decay method used is exponential decay and the decay rate chosen for the cooling schedule-1 is 0.55. When these parameters

are chosen the total cost obtained is 1500/- which is larger than the actual cost. The value of ϵ also converges faster within 12 episodes. Therefore it's implied that proper balancing between exploration and exploitation is not achieved. In cooling schedule-2 the decay rate is increased to 0.85 the energy cost obtained is 1235/-Value of α is kept as 0.01 in both cases. When the value of α is changed to 0.1 the desired result is obtained only for decay rate=0.99. Therefore finding the accurate decay rate is a time-consuming process. It implies that when any one of the parameter is varied it fully affects the performance of the algorithm.

From Fig.4.5 it's clear that the value of ϵ is decayed only after a specific period of time i.e., discrete decay method. When the step size is larger means the number of steps is smaller, the accuracy is lesser. For discrete decay, the number of steps is determined by the episodes and intervals. Even though the dependency of exponential decay on the parameters is more than discrete decay, the convergence time for discrete is less than exponential. Consolidated details of the decay rates and the number of episodes for the convergence are given in the table.

4.4 RL algorithm based on Pursuit Algorithm

From section 4.2 we have seen that finding a suitable cooling schedule is a challenging task. In ϵ-greedy algorithm the value of ϵ decay can only be obtained using the trial and error method. The dependency of parameters on the performance of the algorithm is more. When the value of any one of parameters is varied undesirable results are obtained and it's a time-consuming process. There is an efficient method to balance exploration vs exploitation known as Pursuit algorithm. In pursuit algorithm, there is only one parameter to be used i.e., convergence rate β. Here we investigate the effect of β on convergence. Pursuit algorithm is based on probability distribution. Each action in all the states is assigned with equal probability P(x,a). The action selection is based on the current estimated value of P(x,a). After the action is chosen new state is obtained and the cost incurred for that particular action is calculated using the function g(x_k,a,x_{k+1}). The Q values

are updated using the Bellman equation with the current estimated parameters. For the present state, the greedy action taken is a^*=argmin Q(x',a'). The probability of the state x_k is updated using the following equations.

$$P^{n+1}(x_k, a_k) = \begin{cases} P^n(x_k,a_k)+\beta(1\text{-}P^n(x_k,a_k)), & if a_k=a_g \\ P^n(x_k,a_k)\text{-}\beta(P^n(x_k,a_k)), & if a_k\neq a_g \end{cases} \tag{4.5}$$

From the equation its clearly evident that the probability of taking the greedy action is increased towards 1and the probability of the non-greedy actions are decreased towards 0. Here the probability is dependent only on one parameter that is β. In pursuit algorithm, predefined episodes are not required which makes it easier. Therefore the process will terminate when the probability of the greedy actions in all the states reaches near to 1.

4.4.1 Implementation of RL using Pursuit Algorithm

Algorithm 3: **Overall Algorithm**

Define the environment

Initialize the β, α and penalty.

for 1 to number of flexible loads **do**

Initialize the Q values to zero

Initialize the P values to 0.5 for each state-action pair

Obtain the optimal schedule for a particular load

end for

Algorithm 4: **Optimal Schedule for a particular load**

while (minP<0.999) **do**

Initialize the state vectors [x(1)=s_j and x(2)=0] and count=0

Initialize the convergence rate, learning rate and penalty.

while (x(1)<f and x(2)<l) **do**

Select the random action based on probability distribution

Find the new state using the transition function

Calculate cost using the reward function

Updation of Q values

Obtain the a^*=argmin Q(x',a')

Increase the Probability of a_g using eqn.4.5

$maxP(Count)$=max[P]

Count++

end while

end while

minP=min[maxP]

In Pursuit algorithm as mentioned earlier the probabilities of all the action for a particular state is assigned with equal probabilities. Since there is no requirements of predefined episodes in this algorithm the terminating condition is given in such a way that the probability of optimal action in all the states approaches to unity.

During the scheduling of load-1, the maximum probabilities of all the states are stored in the array maxP. After completing the scheduling of load-1, the probabilities of action with minimum probability (minP) is obtained from the maxP array. The obtained value is compared with converging condition i.e., minP < 0.99. If the condition is not satisfied then the scheduling of load-1 is again repeated until the probability of all the optimal actions converges to unity. The performance of the algorithm for variation in the convergence factor β is also analysed.

4.4.2 Result of load scheduling using pursuit algorithm

Here we consider the case of 5 residential loads. The load details are mentioned in table 4.1 and 4.2

Table 4.10: Parameter considered for pursuit algorithm

Parameters	Values
α	0.1
γ	1
β	0.01

TIME	D1	D2	D3	D4	D5
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	1
8	0	0	0	0	1
9	1	0	0	0	1
10	1	1	0	0	1
11	1	0	0	0	0
12	1	0	0	0	0
13	0	0	0	0	0
14	0	1	1	1	1
15	0	1	1	1	1
16	0	1	1	1	1
17	0	0	1	1	1
18	0	0	1	1	0
19	0	0	0	1	0
20	0	0	0	0	0
21	0	0	0	0	0
22	0	0	0	0	0
23	0	0	0	1	0
24	0	0	0	0	0

Fig. 4.7: Scheduling of all the loads using Pursuit algorithm

Implemented the load scheduling using the pursuit algorithm and the schedule obtained is given in Fig.4.7 and the parameters chosen are given in Table 4.10. Action sequence for each load is given in Table 4.11. From that it is seen that load-1 is turned on during the time period 8,9,10,11.

Table 4.11: Action sequence of each load

Load	**(s,f,l,rt,udc)**	**Action Sequence**
1	(8,15,4,10,20)	8,9,10,11
2	(8,15,4,10,0)	8,9,14,15
3	(11,19,5,7,2)	14,15,16,17,18
4	(10,24,7,5,1)	14,15,16,17,18,19,23
5	(6,18,8,5,3)	6,7,8,9,14,15,16,17

In the earlier case, it is easier to implement the scheduling when the load specifications are known. Now let's analyse the flexibility of the algorithm when the specifications of load are random. For the analysis here we consider 100 loads with random load specifications. The load specifications i.e., (s_j,f_j,l_j,rt_j,udc) are taken as a random values within a specified limit. Value of starting time s_j is taken as a random value with uniform distribution in the range of [1,5]. The value of f_j is taken as random value between [10,15]. Likewise the values of rt_j and udc is chosen in the range [5,20] and [1,10] respectively. The cost of energy considered for the analysis is the same as given in Table 4.2

Table 4.12: Load Specifications

Parameters	Values
s_j	(1,5)
f_j	(10,15)
l_j	(1,5)
rt_j	(5,20)
udc	(1,10)

The load schedule obtained for 5 loads from 100 random loads using the above specification is given in the Table 4.13

Table 4.13: Action Sequence of 5 loads

Load	Specifications	Action Sequence
1	(3,13,8,7,6)	3,4,5,6-10
2	(6,12,5,6,5)	6,7,8,9,10
3	(7,14,6,9,10)	7,8,9,10,11,12
4	(7,16,3,8,9)	7,8,9
5	(6,13,5,20,2)	6,7,8,9,10

If we consider the case of load-1 then its specifications are (3,13,8,7,6) which means that starting time is 3 and the finishing time is 13 hours within that duration we have to schedule the load for 8 hour duration. According to the price at that period of scheduling and udc the action sequence obtained for scheduling the loads are given in Table 4.13.

In the case of load-1 when udc is 6, the load has to be scheduled for a 8 hour duration within 13 hours. In between these periods, the price of electricity is low only during 1-10 hours from 11-13 the price is high. Since the value of udc is high the load is scheduled for the consecutive 8-hour duration. If we consider the case of load-2 its udc is 5, therefore the load is also scheduled for a consecutive 5 hour duration. For load-3, the value of udc is 10, the load has to be scheduled for a consecutive 6 hour duration. If we consider the case of load-4 it has to be scheduled within 7-16 hours with a 3 hour duration. The price of energy is low during the time period 1-10 and 14-19. So the load can be scheduled in between the low energy price periods, but the value of udc is 10. Therefore consumer's comfort is given more preference than cost minimization, the load is scheduled during the period 7,8,9. For load-5 the value of udc is only 2 hence the load can be scheduled in a low energy price period. The scheduling the done during the period 6-10. Likewise, scheduling is done for 100 loads and the total cost obtained by scheduling 100 loads is ₹2095.

4.4.3 Analysing effect of udc

For analyzing the effect of udc, the value of udc is varied from 0-10 for all the loads. Obtained results are shown in Table.4.14.

Table 4.14: Effect of udc

udc	Cost(₹)
0	16470
2	20830
4	21505
6	22580
8	24335
10	24335

From the table, its seen that when the value of udc is increased the cost of energy also gets increased. When udc is high for a load then the scheduling of that load is done consecutive duration irrespective of the cost at that time. Consumers comfort is given more priority compared to cost minimization. Basically, the value of udc can range from 0-10 from the analysis of udc it's seen that cost of energy is almost constant after 8. Therefore the maximum value of udc given to high priority load can be chosen near to 8.

Load scheduling when the tariff is real-time pricing

For price based DR there are various tariff structures as disccussed earlier. Earlier we have implemented the scheduling for ToU. Now lets analyse the adaptability of the algorithm when the tariff structure is changed. In order to implement the scheduling using RTP. Along with the already existing cost a random value of uniform distribution between (-2,2) is added to the cost given in the table 4.15

In order to analyse the schedule the cost obtained for the first few runs are plotted.

Table 4.15: Price of Electricity

Time	Price of Electricity
1-10	5+(-2,2)
11-13	12+(-2,2)
14-19	5+(-2,2)
20-22	10+(-2,2)
23,24	5+(-2,2)

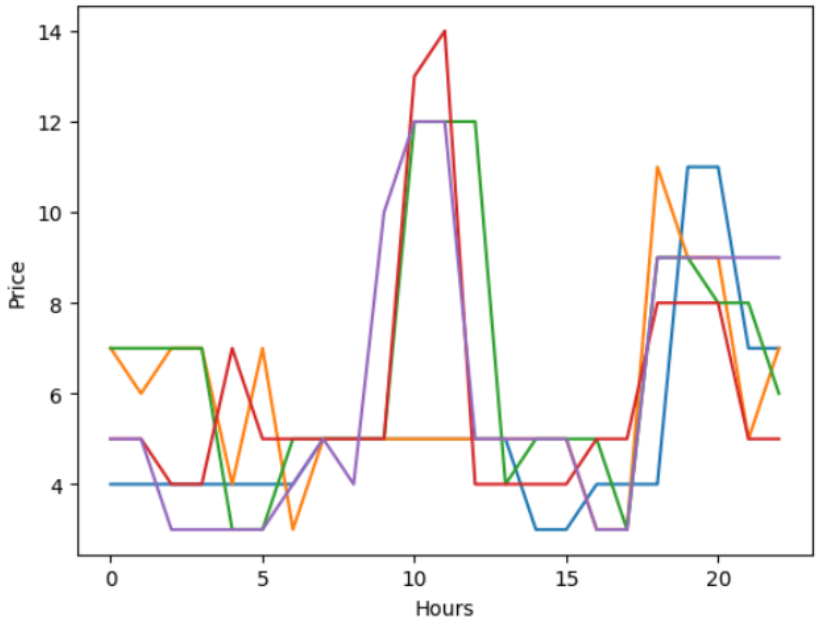

Fig. 4.8: Price of electricity

From the generated cost its seen that the peak period is around 8-12 hours. If the load specifications are known the scheduling should be done during the off peak by satisfying the necessary conditions like the load specification and consumers comfort.

The results of load scheduling obtained after the implementation with the real time time pricing and ToU is same. Since the cost is given random its different for different runs. Therefore it is infered that the algorithm is adaptable for both when the price is aprior known and random.

4.5 Analysis of effect of parameter in Pursuit algorithm

Since the performance of this particular algorithm is affected only by one parameter β. The value of β is varied and the performance of the algorithm is analysed.

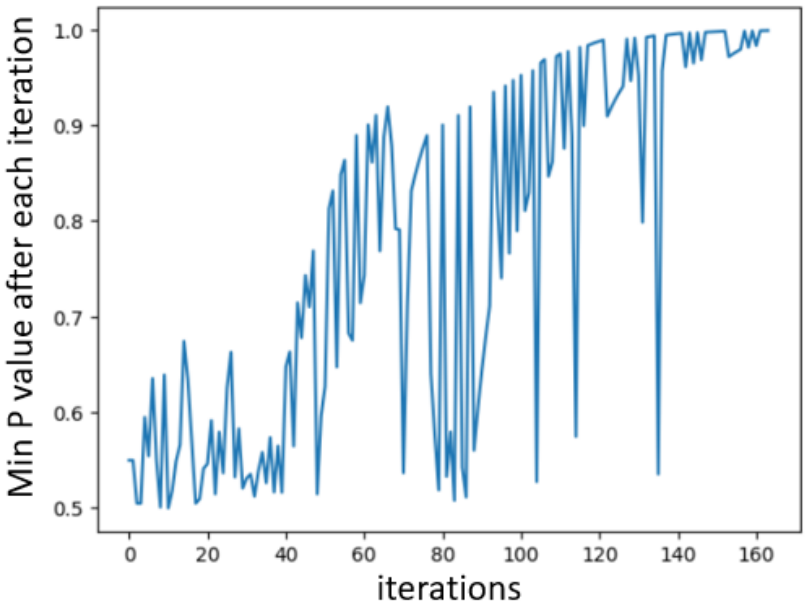

Fig. 4.9: Value of β=0.1 for load-1

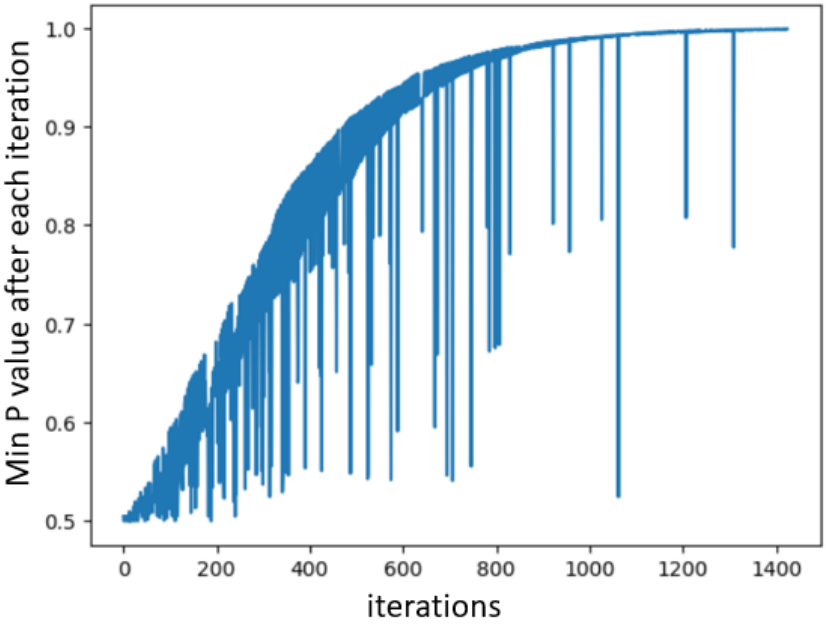

Fig. 4.10: Value of β=0.01 for load-1

Here the algorithm terminates when the probability of the optimal action is close to 1. If we consider a particular load the probability of all the state are considered. As our aim is to make the probabilities of all the optimal actions close to one. For all the loads every state has a specific optimal action. Here for each load the maximum probabilities of each actions are stored among those the one with least value is plotted after each iteration. As we can see when the value of β=1, the value converges to 1. On comparing the above graphs when the value of β is 0.1 the convergence rate is faster. Number of iterations are also very much less which indicates that proper exploration is not done. It means that the probability of all the optimal actions converges faster to 1 within the limited number of iterations. This leads to undesirable results. When the value of β is chosen to 0.01 then its seen the obtained graphs are almost continuous and more accurate on comparing with previous ones as the number of iterations are also greater. Therefore choosing the convergence rate is only the difficulty with pursuit algorithm.

4.6 Conclusion

In this chapter the performance of epsilon greedy algorithm and pursuit algorithm has been analyzed for various conditions. From the analysis its clear that when the scheduling of load is causing more discomfort then the maximum value of udc that can be given is 7. If the range of udc is analyzed with the price of energy then its evident that the maximum value of udc is the average value of energy price.The convergence time of ϵ-greedy algorithm is more for discrete decay method compared to exponential decay method for this scenario. In exponential decay method when the value is chosen near to 0.99 there is proper exploration taking place. It's been obtained using trial and error experimentation which makes it more time consuming. The challenging task of cooling schedule is overcomed using the Pursuit algorithm. Performance of Pursuit algorithm has been analyzed for various values of convergence rate. When convergence rate is smaller the results are more accurate. Predefined episodes are not required as each load takes different iterations for convergence.

Chapter 5

Conclusion

Residential load scheduling is implemented using RL and is simulated in google colab. Algorithm used for implementing the DR is done using Q-learning. The ordinary exploratory policy used for the action selection is ϵ-greedy. Sensitivity of the algorithm to various parameters affects its performance. Obtaining the cooling schedule is a challenging task in this algorithm. The performance of the algorithm for various values of decay methods and udc is analysed. Changing the value of udc over a particular range helps in determining the maximum value of udc that could be used for a load. To overcome the challenges an efficient algorithm Pursuit algorithm is used. The algorithm depends only on one parameter i.e., β. Performance of the algorithm is analysed for various values of convergence rate and udc. Along with that implemented the scheduling of 100 loads. The load specifications are considered as random value with uniform distribution within the specified limits as detailed in Chapter-5. Even though the pursuit algorithm depends only on the convergence rate, obtaining the convergence rate for a specific application is also a challenging task.

The focus of the thesis is on the pursuit algorithm, there are several algorithms that can be used for implementing DR. For the analysis purpose, we considered only the case of specific loads as future scope it can be extended to include various types of loads.

Bibliography

[1] Fayiz Alfaverh M. Denai, and Yichuang Sun,*"Demand Response Strategy Based on Reinforcement Learning and Fuzzy Reasoning for Home Energy Management"*,IEEE Acess,vol. 8,pp.39310– 39321,February 2020.

[2] Luciana Marques,Miguel Heleno,Wadaed Uturbey,*" Transactive control for residential demand-side management: Lessons learned from noncooperative game theory "*,Decentralized framework of power system, pp. 277-317,2022

[3] Jingkai Jia and Wenlin Wang, *"Review of reinforcement learning research"*, 5th Youth Academic Annual Conference of Chinese Association of Automation,2020.

[4] Bizzat Hussain Zaidi, Sarah Sami Khan, Falak Naz Farooqui and Ambreen Abdul Razaque,*"Demand Response for Industrial Facilities"*,IEEE Conf. Transportation Electrification Technology,August 2020.

[5] Donald Azuatalam, Wee-Lih Lee, Frits de Nijs, Ariel Liebman, *"Reinforcement learning for whole-building HVAC control and demand response"*,Elsevier Energy and AI, vol.20, August 2020.

[6] Sangyoon Lee and Dae-Hyun Choi (2019),*"Reinforcement Learning-Based Energy Management of Smart Home with Rooftop Solar Photovoltaic System, Energy Storage System, and Home Appliances"*, Home Automation for the Internet of Things, September 2019. https://doi.org/10.3390/s19183937.

[7] Xuefei Huang,SeungHo Hong,Mengmeng Yu,Yumein Ding,andJunhui

Jiang,*"Demand Response Management for Industrial Facilities: A Deep Reinforcement Learning Approach"*,IEEE AI Technologies in Power System,vol.7,pp-82194 - 82205,June2019

[8] Mian Hu,Jiang-Wen Xiao,Shi-Chang Cui,Yan-Wu Wang , *"Distributed real-time demand response for energy management scheduling in smart grid"*,International Journal of Electrical Power and Energy,Volume 99, Pages 233-245,July 2018

[9] Bo Zeng,Geng Wu,Jianhui Wang,Jianhua Zhang,Ming Zeng, *"Impact of behavior-driven demand response on supply adequacy in smart distribution systems"*,Elsevier applied energy,vol.202,pp.125-137,September 2017.

[10] N.Y.Olawuyi,M.F.Akorede,E.Femi,Ayeni,R.G.Jimoh, *"Real-time demand response algorithm for minimising industrial consumers electricity billing"*,IEEE 3rd International Conference on Electro-Technology for National Development,2017.

[11] Frederik Ruelens,Bert J. Claessens, Stijn Vandael,Bart De Schutter Robert Babuška,Ronnie Belmans,*"Residential Demand Response of Thermostatically Controlled Loads Using Batch Reinforcement Learning"*,IEEE Transactions on Smart Grid ,vol.8,pp.2149 - 2159,September 2017

[12] Maryam H. Shoreha,Pierluigi Sianoa,Miadreza Shafie-khaha, Vincenzo Loia,João P.S. Catalão,*"A survey of industrial applications of Demand Response"*,Elsevier Electrical Power System Research,vol.141,pp-31-49, December 2016.

[13] Ruilong Deng, Zaiyue Yang, Mo-Yuen Chow, and Jiming Chen,*"A Survey on Demand Response in Smart Grids: Mathematical Models and Approaches"*,IEEE Transactions on informatics,vol. 11,no.3, JUNE 2015.

[14] Imthias Ahamed T Parambath, Jasmin E. A , Faisal R. Pazheri, Essam A.Al-Ammar,*"Reinforcement Learning solution to economic dispatch using pursuit algorithm"*, IEEE GCC Conference and Exhibition (GCC), February 19-22, 2011

[15] S.Danish Maqbool,T.P.Imthias Ahamed and N.H.Malik,*"Analysis of Adaptability of Reinforcement Learning Approach"*, IEEE 14th International Multitopic Conference, February 2011.

[16] T.P.Imthias Ahamed,S.Danish Maqbool and N.H. Malik,*"A Reinforcement Learning Approach to Demand Response"*, Proceedings of The EE Centenary Conference, IISc, Bangalore, 15-17 December, 2011.

[17] A.S.Pabla,*"Electric Power Distribution"*, Tata McGraw-Hill Publishing Company Limited, 2012

[18] R.S.Sutton and A.G.Barto,*"Reinforcement Learning : An Introduction"*, MIT Press, Cambridge, MA, 1998.