DYNAMIC LOAD SCHEDULING FOR ENERGY EFFICIENCY IN A MICROGRID

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Ashutosh Nayak

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2018

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Seokcheon Lee, Chair

School of Industrial Engineering

Dr. Shimon Y. Nof

School of Industrial Engineering

Dr. Roshanak Nateghi

School of Industrial Engineering

Dr. John W. Sutherland

Environmental and Ecological Engineering

Approved by:

Dr. Steven Landry

Head of the School Graduate Program

Dedicated to my parents - Mrs. Shashikala Nayak and Mr. Bhoj Kumar Nayak.

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my major advisor, Dr. Seokcheon Lee, for his excellent guidance and support during my PhD study. He has always been very helpful, accommodating and understanding.

I would also like to thank Prof Shimon Nof, for his guidance throughout the course of my stay at Purdue and, support whenever I needed it. I am thankful to Prof. John W. Sutherland and Dr. Fu Zhao for their collaboration on our work and recommendations on my research topics in this dissertation. I also would like to thank Dr. Roshi Nateghi for providing suggestions, help and resource materials whenever I needed them.

I wish to thank all my DC lab members with whom I had productive and fun discussions - MJ, Chris, Yang, Rahul, Pat, Sungbum, Islam, Ho-Yeong, Archit and others.

I wish to thank Prof Manoj Kumar Tiwari who has always supported me since my undergraduate days and continue to support till date. He will always hold a special place for whatever becomes of me.

Finally, I would like to thank family members for their support throughout this journey.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

SYMBOLS

$A[t]$	Electrical energy stored in battery at time t
$a[t]$	Action vector by the players at t
$a_i[t]$	Action by the player i at t
$B[t]$	Electrical energy discharged from battery at time t
B_{max}	Discharge limit from battery at time t
C	Constants
$\bar{c}[t]$	Binary variable for charging or discharging of the battery at time slot t
d_i	index of loads in dynamic list $SL_i[t]$
D_i	Set of loads in dynamic list $SL_i[t]$
E	Peaker power plants
e	Indices for peaker power plants
$Ha[t]$	Total electricity harvest from renewables at time t
$Ha_s[t]$	Electricity harvest from a solar plant at time t
$Ha_w[t]$	Electricity harvest from a wind plant at time t
i	Indices for different consumers
j	Indices for different machines in production line
k	Indices for different production line job types
K_c	Number of clusters in K means clustering
$L[t]$	Total load at time t
$l_{j,k}[t]$	Decision variable for scheduling job k in machine j at time t
$m[t]$	Message vector to consumers from SO at time t
$m_i[t]$	Message to consumer i from SO at time t
$M_i[t]$	Cumulative message to consumer i till time t

N	Total number of consumers
O	Set of incomplete Orders in a job shop manufacturing facility
o	Indices for the incomplete orders
$p_1[t]$	Electricity cost from Microgrid at time t
$p_2[t]$	Electricity cost from Macrogrid at time t
$p_3[t]$	Electricity cost from peaker plants at time t
$p_4[t]$	Electricity selling price from μG at time t
$p_5[t]$	Electricity price from μG_u to μG_v at time t
$Q_i[t]$	Virtual queue for consumer i at time t
Q_i^{CCE}	Virtual queue for consumer participation such that the consumer participates based on information from the SO
Q_i^P	Virtual queue for consumer participation such that the consumer participates collaboratively with other participants and the SO
$Q_d[t]$	Virtual queue for delays in job shop scheduling at time t
$Q_p[t]$	Virtual queue for peak consumption in job shop scheduling at time t
$r[t]$	Request vector from consumers to SO at time t
$r_i[t]$	Request from consumer i to SO at time t
$R_i[t]$	Cumulative request from consumer to SO i till time t
$ReLU$	Rectified Linear Unit t
$S[t]$	Electricity sold from Microgrid to Macrogrid at time t
T	Set of time slots
$t \in \{1, 2, ..., \lvert T \rvert\}$	Indices for time slots t
$\bar{u}_i$	Time average utility of consumer i
$u_i(a_i[t])[t]$	Utility function of player i at time t when she participates ($a_i[t] = m_i[t]$)
$u_i(b_i[t], a_i^{-1}[t])[t]$	Utility function of player i at time t when she participates but $a_i[t] \neq m_i[t]$ and $a_i[t] = b_i[t]$
$u_i(b_i[t], 0)[t]$	Utility function of player i at time t when she does not participate

$G[t]$	Electricity bought from Macrogrid at time t
$X[t]$	Electricity consumed that was harvested from the renewables at time t
$x_{i,r}^{ns}[t]$	Decision variable for time shiftable load at time t
$x_{i,h}^{ps}[t]$	Decision variable for modifiable power load at time t
$Y[t]$	Electrical energy stored in battery at time t
Y_{max}	Charging limit of battery at time t
μ	Microgrid
$\alpha_{j,k}$	Power required when job k is scheduled in machine j at time t
$\alpha_d[t]$	Weight for Virtual queue $Q_d[t]$ at time t
$\alpha_p[t]$	Weight for Virtual queue $Q_p[t]$ at time t
δ	Control variable for order delays
χ	Control variable for peak consumption constraint
$\Delta[t]$	Lyapunov function
$\nabla[t]$	Drift+penalty function

ABBREVIATIONS

ACF	Auto Correlation Function
AIC	Akaike Information Criteria
AR	Auto Regression
ARMA	Auto Regression Moving Average
ANN	Artificial Neural Network
BM	Benchmark
CC	Central Controller
CPP	Critical Peak Pricing
CPS	Cyber Physical Systems
CR	Critical Ratio Heuristic
DL	Deep Learning
DER	Distributed Energy Resources
DLC	Direct Load Control
DOE	Department of Energy
DR	Demand Response
DSM	Demand Side Management
EA	Evolutionary Algorithm
ED	Economic Dispatch
EDD	Earliest Due Date Heuristic
FCFS	First Come First Serve Heuristic
GA	Genetic Algorithm
HVAC	Heating, Ventilation and Air - Conditioning
IBR	Inclining Block Rates
ICT	Information and Communications Technology

IoT	Internet of Things
LO	Lyapunov Optimization
LP	Linear Programming
LSTM	Long Short Term Memory
MA	Moving Average
MDP	Markov Decision Process
ML	Machine Learning
MLP	Multi Layer Perceptron
MILP	Mixed Integer Linear Programming
MINLP	Mixed Integer Non - Linear Programming
NN	Neural Networks
PACF	Partial Auto Correlation Function
PHEV	Plug-in Hybrid Electric Vehicle
PE	Policy Evaluation
PS	Policy Search
PSO	Particle Swarm Optimization
PI	Policy Iteration
RL	Reinforcement Learning
RNN	Recurrent Neural Network
SA	Simulated Annealing
SARSA	State-Action-Reward-State-Action
TOU	Time Of Use
UC	Unit Commitment
PI	Value Iteration
VPP	Variable Peak Pricing

ABSTRACT

Nayak, Ashutosh PhD, Purdue University, December 2018. Dynamic Load Scheduling for Energy Efficiency in a Microgrid. Major Professor: Dr. Seokcheon Lee.

Growing concerns over global warming and increasing fuel costs have pushed the traditional fuel-based centralized electrical grid to the forefront of mounting public pressure. These concerns will only intensify in the future, owing to the growth in electricity demand [1]. Such growths require increased generation of electricity to meet the demand, and this means more carbon footprint from the electrical grid. To meet the growing demand economically by using clean sources of energy, the electrical grid needs significant structural and operational changes to cope with various challenges.

Microgrids (μGs) can be an answer to the structural requirement of the electrical grid. μGs integrate renewables and serve local needs, thereby, reducing line losses and improving resiliency. However, stochastic nature of electricity harvest from renewables makes its integration into the grid challenging. The time varying and intermittent nature of renewables and consumer demand can be mitigated by the use of storages and dynamic load scheduling. Automated dynamic load scheduling constitutes the operational changes that could enable us to achieve energy efficiency in the grid.

The current research works on automated load scheduling primarily focuses on scheduling residential and commercial building loads, while the current research on manufacturing scheduling is based on static approaches with very scarce literature on job shop scheduling. However, residential, commercial and, industrial sector, each contribute to about one-third of the total electricity consumption. A few research works have been done focusing on dynamic scheduling in manufacturing facilities for energy efficiency. In a smart grid scenario, consumers are coupled through electricity pool and storage. Thus, this research investigates the problem of integrating produc-

tion line loads with building loads for optimal scheduling to reduce the total electricity cost in a μG.

This research focuses on integrating the different types of loads from different types of consumers using automated dynamic load scheduling framework for sequential decision making. After building a deterministic model to be used as a benchmark, dynamic load scheduling models are constructed. Firstly, an intelligent algorithm is developed for load scheduling from a consumer's perspective. Secondly, load scheduling model is developed based on central grid controller's perspective. And finally, a reinforcement learning model is developed for improved load scheduling by sharing among multiple μGs. The performance of the algorithms is compared against different well-known individualistic strategies, static strategies and, optimal benchmark solutions. The proposed dynamic load scheduling framework is model free with minimum assumptions and it outperforms the different well-known heuristics and static strategies while obtains solutions comparable to the optimal benchmark solution.

The future electrical grid is envisioned to be an interconnected network of μGs. In addition to the automated load scheduling in a μG, coordination among μGs by demand and capacity sharing can also be used to mitigate stochastic nature of supply and demand in an electrical grid. In this research, demand and resource sharing among μGs is proposed to leverage the interaction between the different μGs for developing load scheduling policy based on reinforcement learning.

1. INTRODUCTION

1.1 Electrical grid in transformation

US electrical grid is the biggest and the most complex machine ever built serving 318.9 million people. There are about 7,700 power plants that produce and distribute electricity to homes, businesses and other consumers as shown in Figure 1.1. That electricity travels through more than 160,000 miles of high-voltage electric transmission lines that reach into every nook and cranny of the country [2]. The electric grid also accounted for approximately 37% of energy related carbon dioxide emissions with coal fired and natural gas fired power plants accounting for 71% and 28% respectively [1].

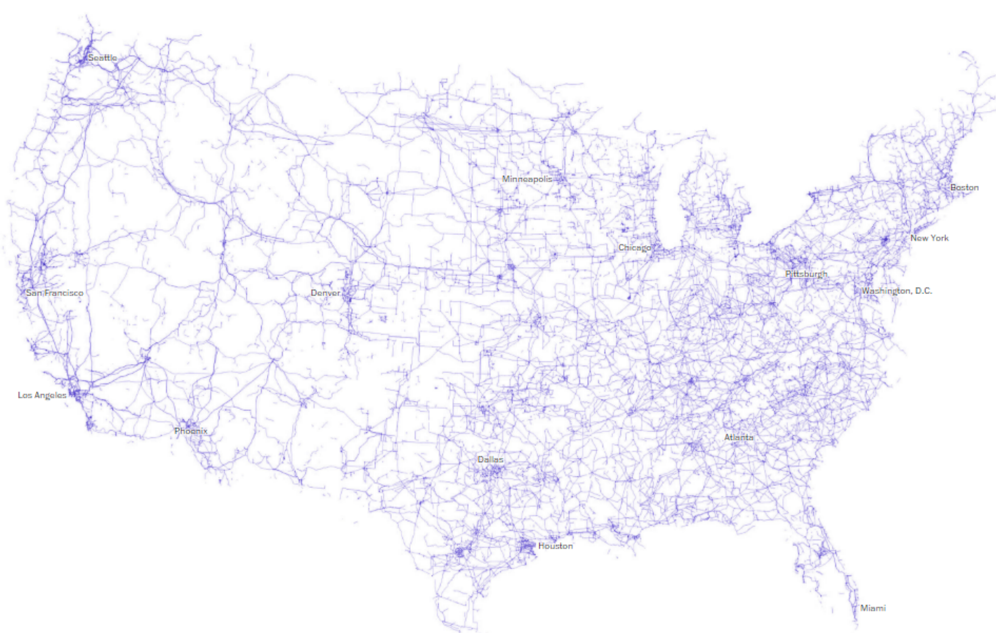


Fig. 1.1. The USA electrical grid

The fundamental architecture of todays electricity grid is based on the idea of a top-down radial transmission system predicated on unidirectional energy flows from large centralized plants to the consumers [3]. Most of these power plants are fuel based that use coal and natural gas as shown in Figure 1.2.

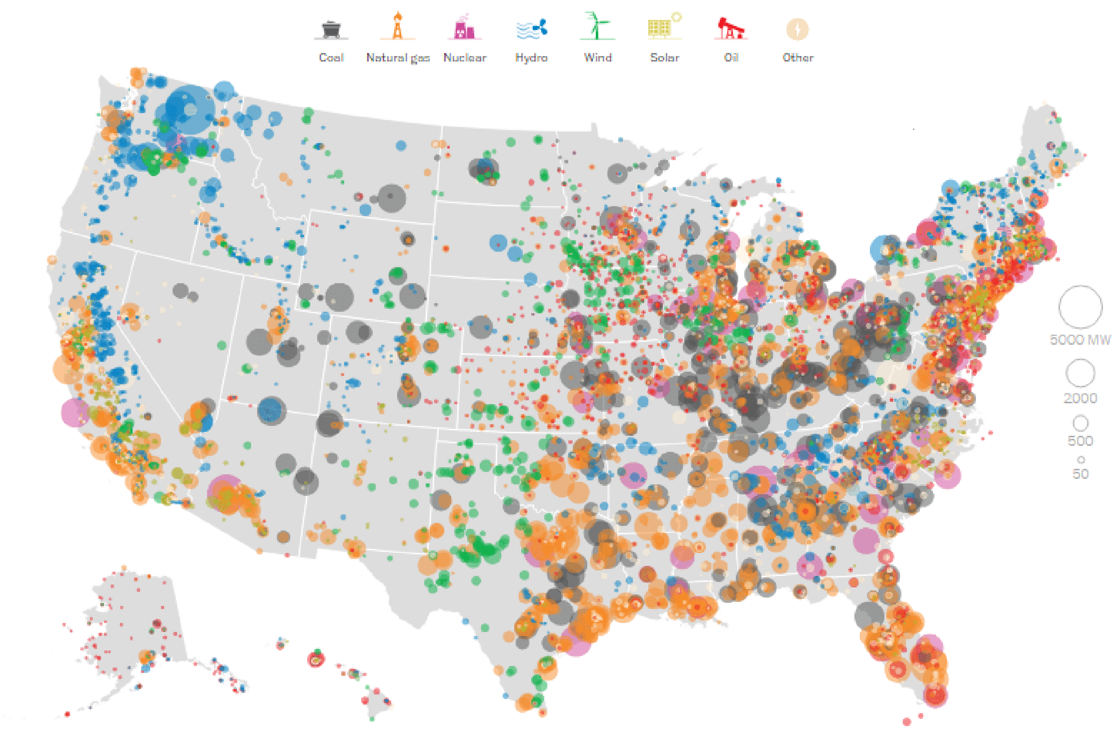


Fig. 1.2. Fuel based power plants across USA

This unidirectional flows have led to construction of excess generation capacity leading to increasing CO_2 emissions and consumer's electricity cost. These concerns will only intensify in the future, owing to growths in demand [1]. Such growths require increased generation of electricity to meet the demand, and this means more carbon footprint from the electrical grid. There is also a concern that increasing supply cannot meet the growing demand by creating additional capacity as demand is expected to grow as shown in Figure 1.3. The estimated cost to upgrade the electric grid to meet the peak demand is $180 billion and electricity prices are expected to rise by 2.8% [4].

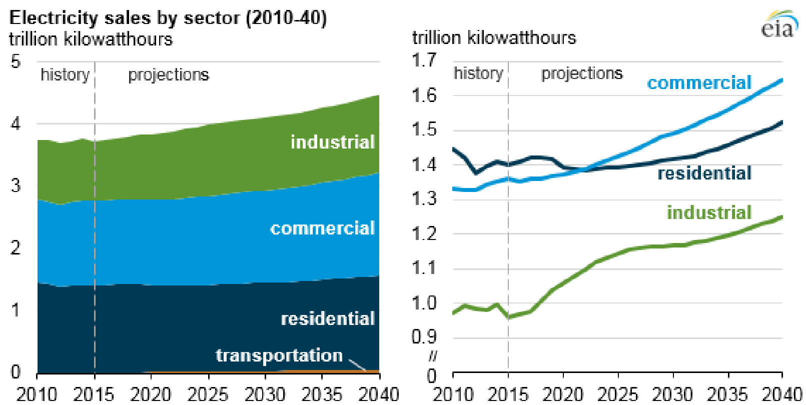


Fig. 1.3. Growing electricity demand

This concerns stem from the fact that the current electricity grid is aging, inefficient and becoming obsolete. This makes efficient use of electricity critical. Two statements that confirms the inefficiency of electric grid are:

- The current electric grid has been rated D+ by American Society of Civil Engineers
- A statistic by EIA shows for every 1 MW of power consumed by U.S. commercial and residential customers, rate payers are paying for 2.2 MW of generation and transmission capacity

These growing concerns over global warming and increasing fuel cost has pushed the traditional fuel-based centralized electrical grid to the forefront of mounting public pressure. The electric grid needs transformation and is undergoing the process of transformation. Two major drivers of this transformation are growing digitization and improved technology for integration of renewables. A report by Ernst and Young mentioned that US$500 B worth of investment in digital grid is expected in the next 5 to 7 years [5]. On one hand, digitization enables in getting real time and accurate information about electricity consumption and generation, running devices at optimal time and automate the system of meeting electrical demands. On the other hand, improved technology has reduced the electricity cost from renewables and integration of renewables provide us a clean and cheap alternative to expensive and carbon intensive electricity from fuel based power plants.

The two major drivers for electric grid transformation can be leveraged by structural and operational level changes in the grid. Line losses are a major contributor of inefficiency in the grid. Losses from distribution lines are estimated to be \$19.5 billion by 2025. Generating electricity using renewables close to the consumption point would reduce line loss and build greater resiliency. This local generation while integrating renewables led to the development of the concept of μGs. Australia plans to shift to shift to a robust 100% renewable grid. Google plans to be powered 100% by renewables by 2017. Future electrical grid is envisioned to have high penetration of μGs with the electrical grid as a network of interconnected μGs.

However, the problem of supply and demand matching is not simple in the case of electricity because of the stochastic nature of electricity harvest from renewables and uncertain consumer demand. A real time load serving platform will make decisions dynamically so that the overall efficiency in the grid could be achieved. This research is thus motivated by the growing interest in μGs to develop a framework for automated load scheduling platform that schedules electrical loads dynamically based on the current information.

Moreover, dynamic automated load scheduling could help in achieving energy efficiency in μGs. Automated load scheduling framework could integrate the demands of different consumers and make optimal load scheduling decisions based on global information. μGs are also important for remote places and countries with poor electricity infrastructure. μGs could leverage the electricity generated locally without using distribution line from far-off places, thus, creating a scope for technology skip.

1.2 Smart microgrid

A Microgrid (μGs) is an electrical grid on a smaller scale with defined electrical boundaries. United States DOE defines a μG as: *"A microgrid is a group of interconnected loads and distributed energy resources within clearly defined electrical boundaries that act as a single controllable entity with respect to the grid. A microgrid can connect and disconnect from the grid to enable it to operate in both grid-connected and island-mode"* [6]. Smart μG is analogous to the smart grid with information sharing, connectivity, coordination and capability of intelligent decisions.

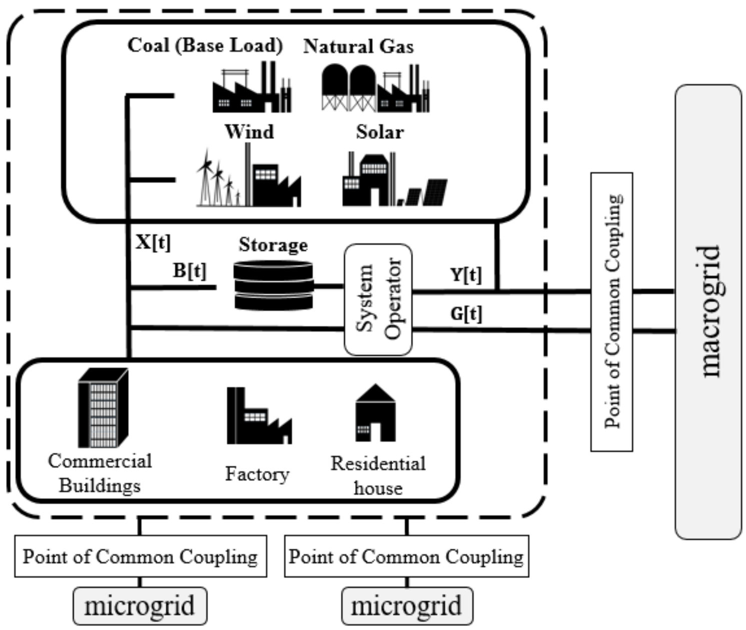


Fig. 1.4. Schematic representation of a μG

Schematic representation of a μG is shown in Figure 1.4 that shows how a μG interacts with macrogrid and other μGs. The characteristics of μGs as described in [7] are:

- Ability to operate independently or in cooperation with macrogrid. When μG need electricity, it connects to macrogrid and disconnects under different unfavorable circumstances such as black out/ brown out events.

- μGs have defined electrical boundaries and serve local demand.
- Ability to integrate distributed energy resources and storage for clean energy
- Ability to achieve high system performance and efficiency through load control

As mentioned in Section 1.1, the electrical grid based on large centralized power plants is highly inefficient and increasingly becoming obsolete. μGs can provide an answer to this inefficiency as they serve local needs, thereby, reducing line losses and building resiliency. μGs are better equipped to take advantages of smart grid technologies. Also, the defined electrical boundaries of a μG enables optimal load scheduling to achieve energy efficiency. The major advantages of μG are:

- μGs help in reduction of distribution losses since electricity is generated close to its end users.
- μGs strengthen reliability by a bottom-up rather than a top-down approach. Lawrence Berkeley National Laboratory statistics show that 80 to 90% of all grid failures begin at the distribution level of electricity service [6].
- μGs can integrate distributed energy resources for cheap and clean energy, thereby enabling us to achieve energy efficiency. It can also integrate electricity from consumers into the grid.
- Defined electrical boundaries of μG make it tractable and practically feasible to implement real-time demand response programs.

The concept of μGs is becoming more popular because of the advantages provided by μGs e.g. higher reliability, lower operating cost, lower distribution losses, tractable supply chain and other extraneous factors like lowering cost of solar panels, increased fuel costs, investment in storage options, incentives by governmental organizations and increasing penetration of Internet technology [6]. The future grid is envisioned to be an interconnected network of μGs [8]. The concept of μGs would gradually become a strong and effective support of the electrical grid system [9]. Navigant

Research ([10]) provided a projection of the market capacity of μGs as shown in Figure 1.5. Navigant Research also provided a projection of installed capacity of μGs as shown in Figure 1.6.

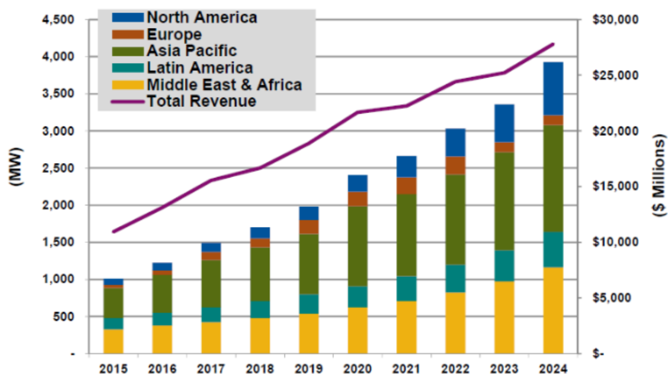


Fig. 1.5. Projection of μGs market capacity

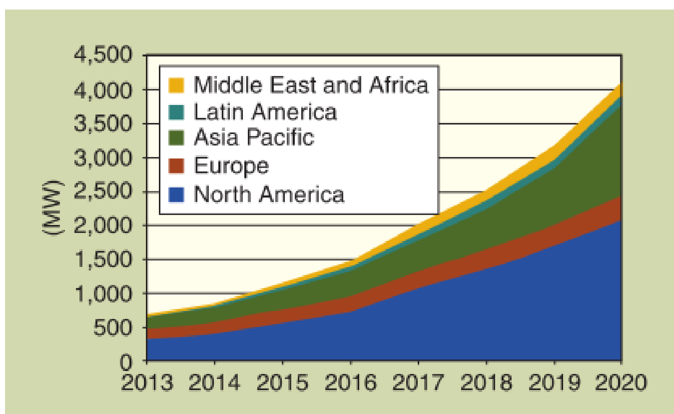


Fig. 1.6. Projection of installed μG capacity

The objective of an electrical grid is to match electricity demand with supply. The supply side and demand side of the μGs are discussed in Section 1.2.1 and Section 1.2.2 respectively. Supply side explains the different sources of electrical energy in a μG and demand side explains the different load types of a consumer which needs to be served.

1.2.1 Supply side

A μG integrates renewables into the distribution system for harvesting clean energy. The renewable sources of energy could be wind farms, solar farms, geothermal stations or tidal energy. Recently, many areas, even countries have run entirely on renewables for a day or month. For example Costa Rica ran on wind energy for 2 months [11] and Tesla runs an entire island from solar power [12]. This shows that if we have enough monitoring and controlling capacity, electrical grid could be entirely run on renewable sources of energy. Annual energy outlook 2015 by US Energy Information projects electricity generation by fuel types as shown in Figure 1.7.

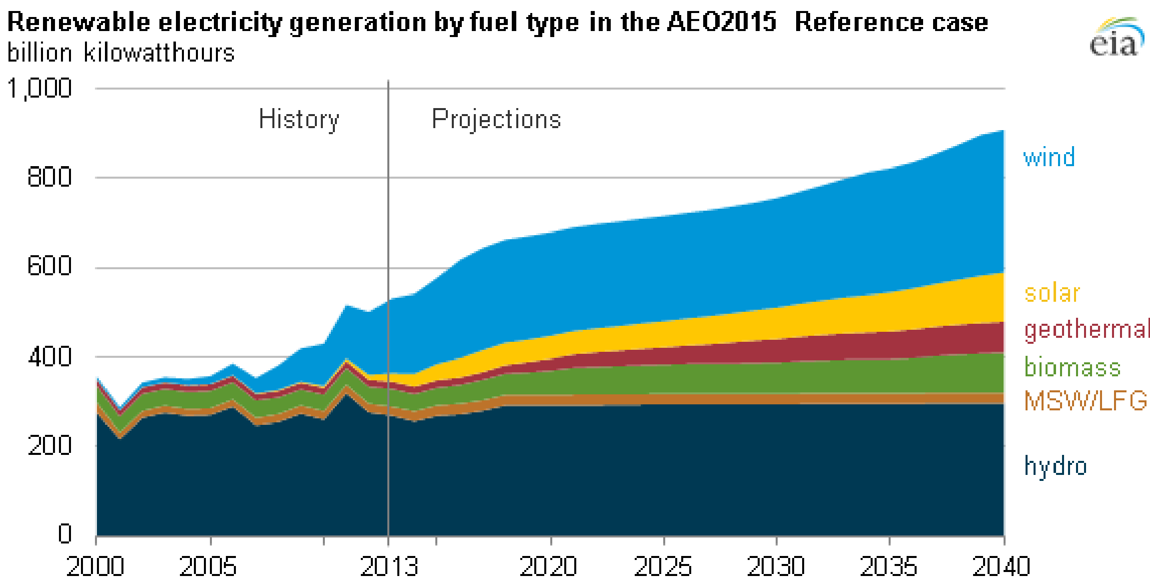


Fig. 1.7. Electricity generation by fuel type

In this research, we consider that the μGs have solar and wind farms. We assume that hourly harvest from these resources are independent draws from a normal distribution with corresponding hourly mean as shown in Figure 1.8 and 1.9 for wind farm and solar farm respectively. The standard deviation is assumed to be 20% of the mean.

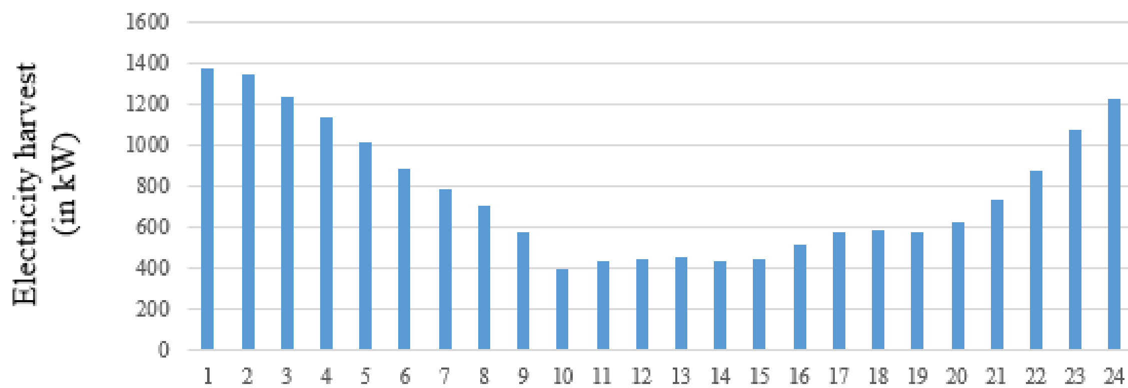


Fig. 1.8. Mean hourly harvest from a wind farm

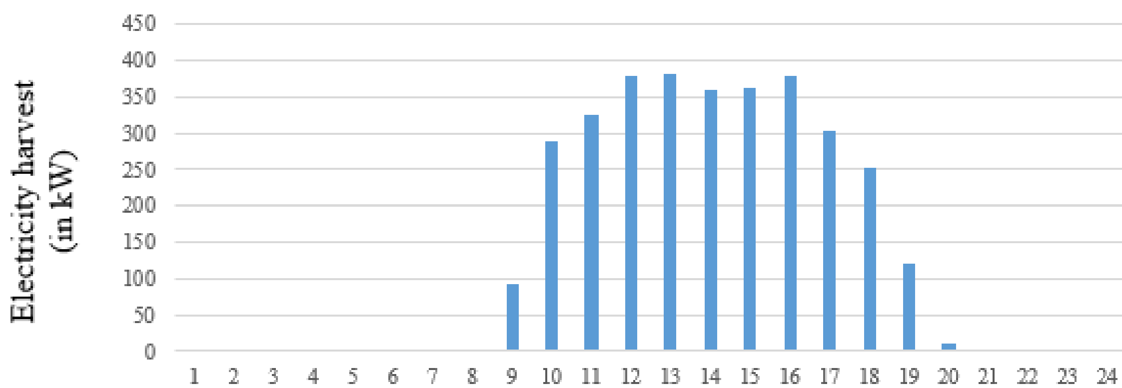


Fig. 1.9. Mean hourly harvest from a solar farm

Biggest challenge to integration of renewables into the grid is their time varying and intermittent nature. Another challenge is voltage synchronization at the switching stations. However, in this research we consider direct current μGs assuming that alternate currents synchronization has been taken care of at the switching stations. Storages can be used to mitigate the stochasticity of electricity harvest from renewables. Storages can smooth out the fluctuations and provide flexibility in shifting electrical energy in time and space. Storages can also help in reducing peak load profile as energy stored in storage could be used at the time of peak load time instead of running carbon intensive natural gas based peaking power plants as shown in Figure 1.10.

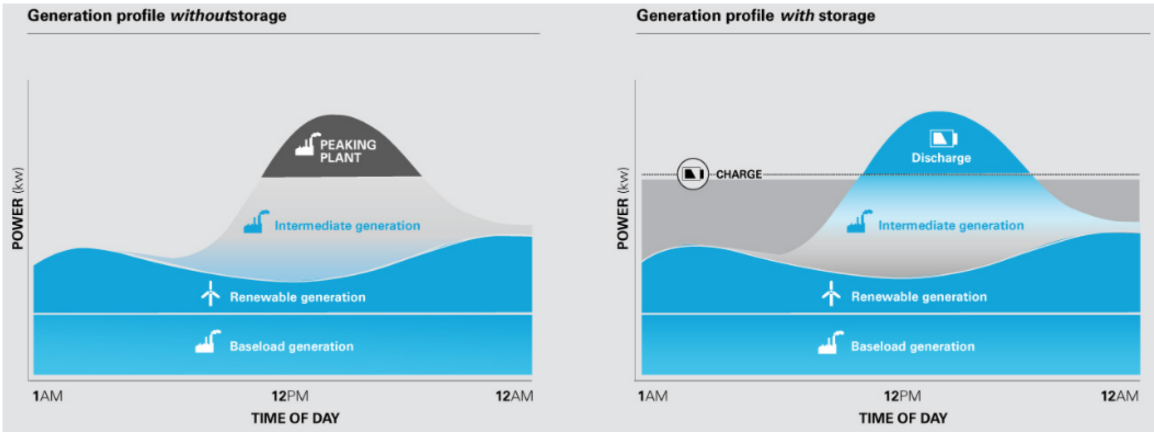


Fig. 1.10. Using storage for peak load reduction

Storages are generally expensive to build and maintain. There are different electrical energy storage options. A brief introduction to different electrical energy storage options is provided in this section. They have been discussed in [13] and [14]. Figure 1.11 shows how the deployment of energy storage systems has increased and is projected to increase. In this research, it is assumed that there is a battery storage. Sensitivity analysis is conducted on different battery capacities.

Battery storage has been used extensively in our day to day life such as in mobile phones and computers. In batteries, electrical energy is stored as chemical energy and discharged when required. Major types of battery storages include lithium ion batteries, sodium sulphur battery, nickel based battery, zinc based battery, hydrogen fuel cells and, lead acid batteries. More advancements is underway for developing batteries with enormous storage capacity.

Hydroelectric storage plants pump water to higher the dam when there is excess electricity in the grid. During peak demand period, water is released through the gates

Fig. 1.11. Projected increase in energy storage systems

to run turbines which generate electricity. Similarly, compressed air energy storage compresses air during off-peak period which could be used to generate electricity during the peak hours.

Electrochemical capacitors perform similar to the lithium ion batteries that store electrical energy in two series capacitor of the electric double layer. Thermal energy storage stores electrical energy in the form of heat energy. For example, heat is stored in the heat sink during off-peak period which could provide heat during the peak periods. It has been deployed mostly at community level. Energy can also be stored in molten salt as heat sink. Flywheel energy storage stores electrical energy in the form of kinetic energy. It is based on spinning a weighted mass at the end of a shaft for generating electricity.

Electric vehicles can also be used as an energy storage that can be charged during the off-peak hours and discharged during the peak hours to minimize the electricity cost. This technology is gaining popularity as the demand for electric vehicles is growing. The growth of different storage technologies in terms of storage capacity is shown in Figure 1.12.

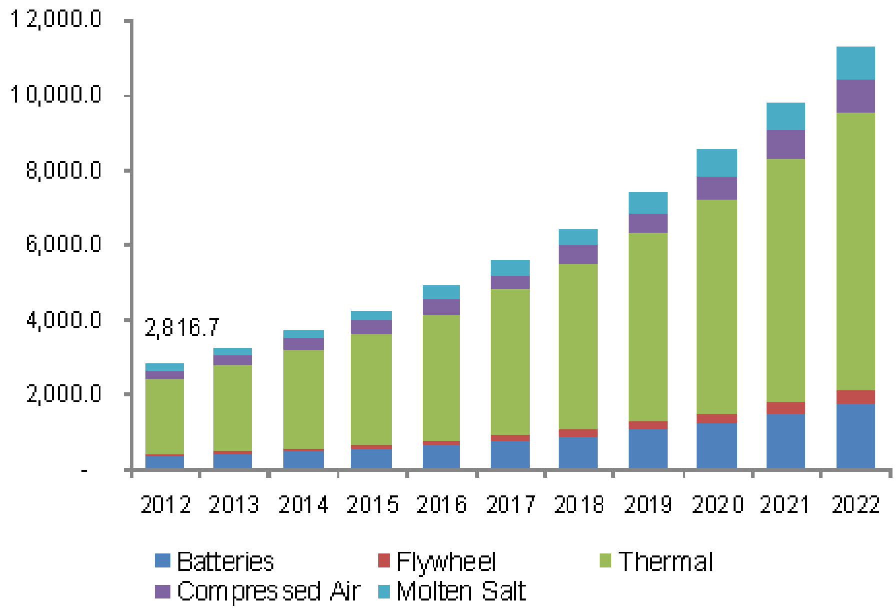


Fig. 1.12. Energy storage systems by market capacity in MW

μGs can buy electricity from the Macrogrid when required. Electricity is drawn through points of common coupling as shown in Figure 1.4. A μG is connected to other μGs and through points of common coupling. If the demand cannot be met by the energy resources of the μG, electricity is bought from the Macrogrid or other μGs based on the contract. In this research, it is assumed that a μG pays Macrogrid based on TOU tariff and the TOU tariff of electricity from Macrogrid is higher than that of μG.

With increasing penetration of renewables such as solar roofs, electric vehicles and, wind mills, even consumers can provide electricity to the μG when required ([15] and [16]). For example, electrical energy stored in electric vehicles can be sold back to the grid at a time when electricity is expensive. A factory can sell the excess harvest to the grid from the wind mill installed at the factory location. Selling back electricity requires two directional electricity flow which has its own technical challenges.

1.2.2 Demand side

Demand side in a μG consists of consumer loads. In this research, we consider that a consumer may be a residential house, a commercial building or a manufacturing facility. All the consumer types have different types of electrical loads. Some loads can be shifted in time and some cannot, thus are called shiftable and flexible loads. There are some industrial loads, particularly, production line loads that are characteristically different from other load types [17]. Definition of different load types is provided in [18].

Definition 1. Non-shiftable loads/ base loads
Non-shiftable loads cannot in time and have to be served when requested. Examples of non-shiftable loads include refrigerator, medical devices in hospitals and, clean room in semiconductor factory. Non-shiftable loads cannot be known exactly and it is estimated for different time slots for every consumer. For a consumer i, the non-shiftable loads are represented as $T_i^{ns}[t] = L_i^{ns}[t]$.

Definition 2. Time-shiftable loads
Time shiftable loads are flexible loads as they can be shifted in time. Time shiftable loads have a minimum start time and the objective is to minimize the waiting time for its operation [19]. They also have latest finish time by which it should be served. The earliest start time and latest finish time form the time window constraints for the time shiftable loads. Examples of time shiftable loads include washing machine, floor cleaning machines and, a job processing machine. Time shiftable loads are uninterruptible loads that run at constant power. For a consumer i, r^{th} time shiftable load is expressed as a 4-tuple $T_{i,r}^{ts} = \{t_{i,r}^{ts,min}, t_{i,r}^{ts,max}, L_{i,r}^{ts}, tr_{i,r}\}$ where $t_{i,r}^{ts,min}$ is the earliest start time, $t_{i,r}^{ts,max}$ is the latest finish time, $L_{i,r}^{ts}$ is the power demand and, $tr_{i,r}$ is the time duration for which the load is to be run. In this research, since we consider discrete time models, $tr_{i,r}$ is expressed in number of time slots. Time shiftable loads are associated with decision variables $x_{i,r}^{ts}[t]$ where $x_{i,r}^{ts}[t] = 1$ if the load is running at time slot indexed as t and $x_{i,r}^{ts}[t] \in \{0, 1\}$.

Definition 3. Modifiable power loads

Modifiable power loads are flexible loads as they can be served at different power levels. Modifiable power loads have time window constraints such that the total energy requirements should be met within their time window. Since energy is equal to power times time, total energy requirement can be met by supplying varying power to the load. Modifiable power loads are interruptible. bhosale et.al ([20]) used the term power shiftable loads for modifiable power loads. Examples of modifiable power loads include PHEV, Heating and, Air conditioner. For a consumer i, h^{th} modifiable power load is expressed as a 5-tuple $T_{i,h}^{ps} = \{t_{i,h}^{ps,min}, t_{i,h}^{ps,max}, L_{i,h}^{ps}, L_{i,h}^{ps,min}, L_{i,h}^{ps,max}\}$ where $t_{i,h}^{ps,min}$ is the earliest start time, $t_{i,h}^{ps,max}$ is the latest finish time, $L_{i,h}^{ps}$ is the total energy demand, $L_{i,h}^{ps,max}$ is the minimum power supply demand and, $t_{i,h}^{ps,max}$ is the maximum power supply demand. Modifiable power loads are associated with decision variables $x_{i,h}^{ps}[t]$ where $x_{i,r}^{ts}[t] \in R^{+}$.

Definition 4. Production line loads

Production line loads are involved in production operations e.g. machining, assembly and processing. These loads are different from other loads because of their properties e.g. sequential completion of jobs, state change cost such as switching a machine state from on to off, machine selection flexibility of jobs, flexibility to change processing speed and, uniterruptability. Different manufacturing set-ups have different arrangement of their production line and the properties of production line loads change with the manufacturing set-up. In this research, we consider that loads ancillary to production line e.g. transportation are included in hourly estimate of non-shiftable loads. Some industrial loads like HVAC, PHEV in parking or clean room have been accounted for in different load types discussed above. Decision variables associated with production line loads include job sequence, job selection, machine selection, processing speed selection and assignment of these decision variables over time. Integrating production line loads with other consumer and load types is one of the major theme of this research.

The mean hourly demand for the non-shiftable loads can be estimated as its time average remain close to constant over long period. It is assumed the distribution of hourly demand is normally distributed with mean shown in the following data and standard deviation is 30% of the mean.

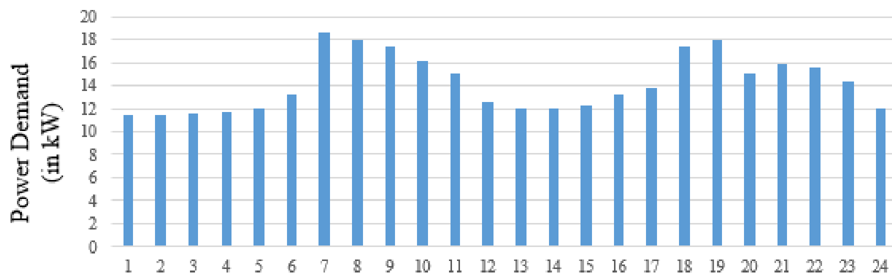


Fig. 1.13. Mean hourly demand for non-shiftable loads: residential house

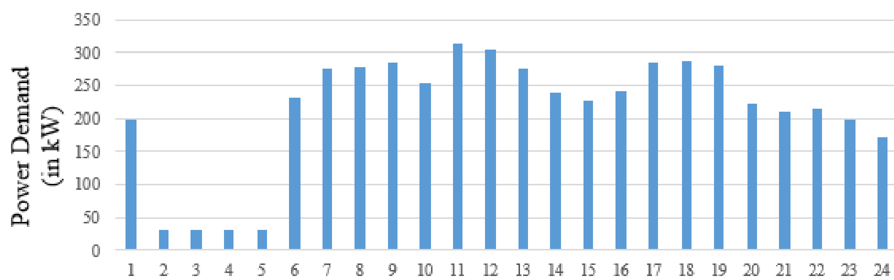


Fig. 1.14. Mean hourly demand for non-shiftable loads: commercial building

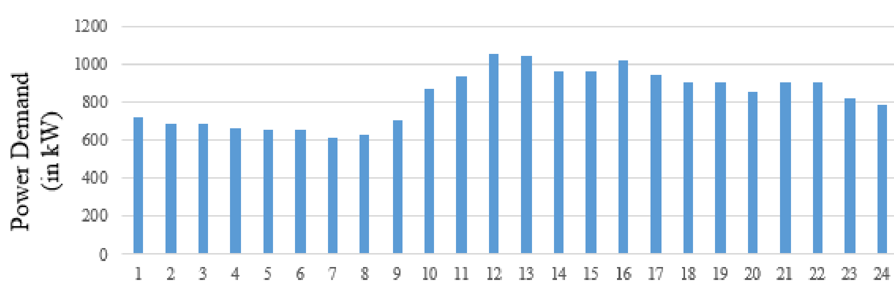


Fig. 1.15. Mean hourly demand for non-shiftable loads: factory

1.3 Automated load scheduling

Energy efficiency is based on real-time demand and supply matching of electricity. Traditionally, supply is managed based on the predicted demand. However, energy efficiency could be achieved by optimal load scheduling. With growing electricity demand, access to electricity in remote areas, smart devices and integration of renewables into the grid, the problem of energy efficient load scheduling becomes very complex and practically impossible to solve using predictive modeling. Moreover, dynamic changes and stochasticity in the system adds complexity and dimensionality to the problem. Nonetheless, the increased complexity provides a motivation for automated load scheduling.

Automated load scheduling is defined as autonomous answer by a central controller called System Operator (SO) to the following question - at any given time, how much load of which consumer is served? The given problem is a scheduling problem as it can be seen as a time indexed assignment problem where loads are assigned to different times based on the available resources. Since this assignment is very complex due to the size and stochasticity, an autonomous controller solves the assignment using global information such that fairness is maintained in the system. This research proposes a framework for automated load scheduling. Automated load scheduling has certain prerequisites. The Major prerequisites include Internet connectivity and smart devices. Internet connectivity is required for real-time information sharing and smart devices are required for responding to messages from controllers. Participation of consumers is also required as scheduling decisions are made based on global information. Figure 1.16 shows the major application areas of IoT. The figure shows that the number of intelligent and ICT enabled smart devices e.g. home appliances, industrial machines and smart meters that comply with the objectives of automated load scheduling are becoming popular and being deployed increasingly. This shows that the proposed automated load scheduling framework is critical for future electrical grid infrastructure.

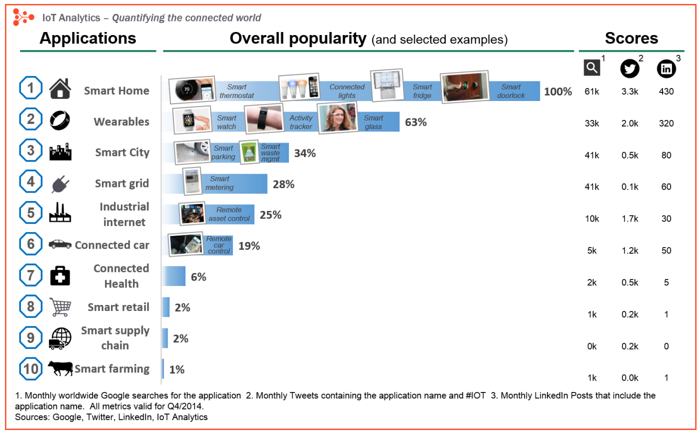


Fig. 1.16. Most popular IoT applications

Automated load scheduling has also drawn interest of the researchers. The interaction between DLCs and SO studied in this research have been studied with different names. Song et. al [21] considered information exchange between DLCs and utility. DLC devices have been studied as energy boxes [22] and energy Macro hubs [23]. Mohsenian et.al [19] studied load scheduling for residential houses through an energy scheduler. Amir et. al [24] considered energy consumption scheduler for autonomous load scheduling. Hu et. al [25] introduced home energy management systems for load shifting. Load management in smart homes by retail electricity provider is studied in Vasinrani et. al [18]. Bhosale et. al [20] used the term controller unit for optimal load shedding with residential consumers. Saad et.al [26] studied communication infrastructure for autonomous load scheduling. Energy management system [27] and energy hub management system [23] performed automatic load scheduling in their research. Huang et. al [28] used central controller for energy sharing. Wang et. al [29] studied automated load scheduling as interaction between consumer, power grid operator and electricity distributor. Intelligent energy scheduler was in charge of electricity distribution among energy users [30]. Siemens has developed an advanced μG management software to dynamically control distributed power generation as per their press release in February 2015 [31].

Automated load scheduling has several advantages as:

- Electricity price changes with time. A consumer need not worry about changing electricity prices as the framework ensures overall minimum electricity cost in the μG whose benefits are transferred to the consumers. it will also help in avoiding rebound effect caused by scheduling multiple loads at the time of minimum cost.

- Information sharing is real-time, thus scheduling decisions can be made based on global information from the electricity demand side and supply side without human intervention.

- Scheduling of different loads is accounted for by the framework. This could enable customized analysis of electricity consumption for the consumers. For example, a consumer can be provided information on total electricity usage by their HVAC.

Privacy is a major concern for consumers. A schematic representation of the automated load scheduling framework is shown in Figure 1.17. The arrows represent the flow of information and the numbers in black circle represent the sequence of information flow. Automated load scheduling is performed by controllers at two levels - DLCs installed at consumers' location and SO at the μG monitoring center. DLCs are connected to different devices and control when to run these devices. The consumers interact with SO through DLCs. Thus, a consumer actually requests demand to the DLC which sends a request to SO in every time slot. SO do not have full information of the different loads. After receiving message from SO, a DLC runs the devices to serve the requests by the consumer. These request could be given by the consumers to DLCs either for the next day or dynamically throughout the day. Thus, the proposed framework ensures privacy of the consumers as consumer information does not go beyond her/his DLC.

The automated load scheduling is modeled as a dynamic scheduling problem with N participants and one manager. DLCs represent the participants and SO represent the manager in dynamic scheduling framework. The dynamic scheduling problem is

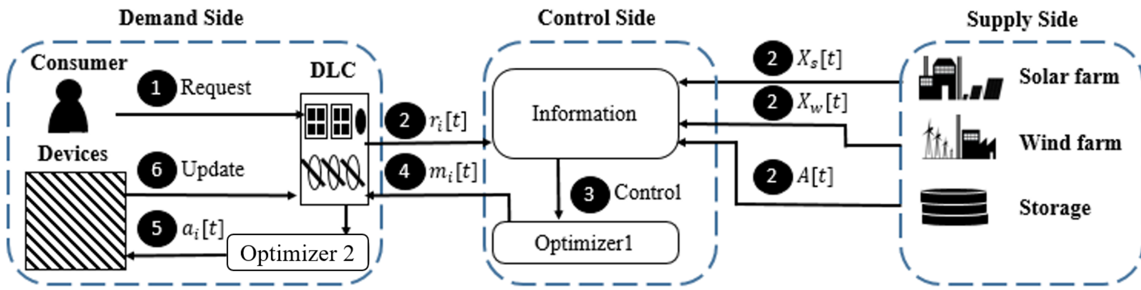


Fig. 1.17. Schematic representation of automated load scheduling

repeated over a sequence of time slots $t \in \{0, 1, 2, ..., T\}$ and there are random events in every time slot. The random events include electricity harvest from renewables and dynamic requests from consumers($\{r_1[t], r_2[t], ..., r_N[t]\}$). The manager observes the random event vector ($\{w_o[t], r_1[t], r_2[t], ..., r_N[t]\}$) but the participants observe only their respective random event ($w_i[t]$). This also ensures privacy of the consumers as the information of one consumer is not revealed to other consumers. The objective of the manager is to maximize the utility function $\sum_{t=1}^{T} \phi(u_i, u_2, ..., u_N)$. The utility function may take different forms based on the objective.

In every time slot t, a participant i sends request $r_i[t]$ to the manager. Thus, the manager observes the request vector $\{r_1[t], r_2[t], ..., r_N[t]\}$. The manager runs Optimizer1 to find optimal message vector $\{m_1[t], m_2[t], ..., m_N[t]\}$. Based on the message vector, a participant performs an action $a_i[t]$. As assumed that the participants participate in the game, $a_i[t] \leq m_i[t], \forall i \in I \, \forall t \in T$. The action message $a_i[t]$ determines which load is served by DLC of player i based on result from Optimizer2.

One of the major concerns with automated load scheduling framework (or with any connected system) is security threat. For this research, it is assumed that the system is secure. We also assume that the Internet connectivity is uninterrupted for smooth flow of information and control of smart devices.

1.4 Reinforcement Learning

Reinforcement learning and machine learning are at the forefront of Artificial Intelligence. With increased computational power, recent successes, advancements and research interest in AI, the question of *Singularity* is looming. Conceptually, reinforcement learning mimics the nature as to how a person learns the world by exploring different options based on the rewards they get. Reinforcement learning consists of agents, environment, system dynamics and rewards. The objective of an agent it to maximize the cumulative rewards they get till the end of the game life.

Recently, Deep reinforcement learning method has achieved human level expertise in certain games like Atari and AlphaGo. Artificial system by Google's Deep mind defeated the World champion Lee Sudol in the game of Go. A large number of big companies have started an Artificial Intelligence arm, that aims at building different applications of reinforcement learning. There is some research in reinforcement learning in smart grid, but it is limited and solves problems with small action space.

Reinforcement learning aims at building computational and mathematical models with the goal of solving computational problems. It has shown connections to the science of Psychology and neurosciences [32]. In their textbook, Sutton and Barto explained the relationship between the mathematical models used in modern reinforcement with the traditional models developed by psychologists and neuroscientists. An famous example is Pavlov's experiments that maps states to rewards, thus making the dog to perform according to the maximum reward without actually understanding the outside stimulus.

The schematic representation of the steps involved in reinforcement learning is shown in Figure1.18. The agent explores the environment to learn the unknown rewards and system dynamics. After learning the environment, the agent takes decisions such that the cumulative rewards is maximized. We use the concept of reinforcement learning and machine learning in this research to convert reinforcement learning problem into a supervised machine learning problem for dynamic load scheduling.

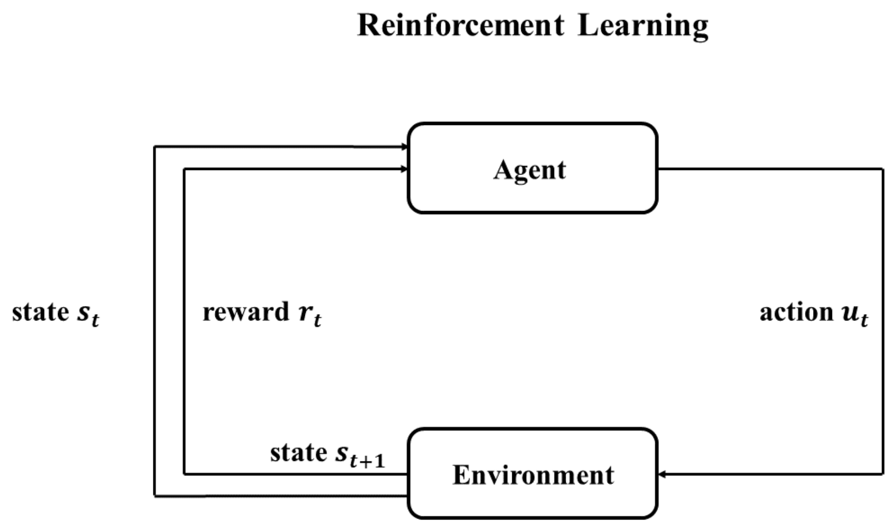


Fig. 1.18. Reinforcement learning iterative procedure

While the reinforcement learning has achieved significant progress and success, the applications of modern reinforcement learning techniques has been limited to games and toy robotic problems. Traditional reinforcement learning methods with policy and value iteration techniques have been applied in real problems but the scope of those methods is limited as it can be used only for problems with small action or state space. In this research, we used modern reinforcement learning techniques to develop a framework for the complex problem of dynamic load scheduling. The problem is complex because of the size of the problem, the size of the action and state space, partial observability of the states and underlying uncertainties. A model free method is developed in this research that considers minimum assumptions about the distribution of stochastic variables.

1.5 Research problem and contribution

This research aims at developing a framework to achieve electrical energy efficiency in a μG using dynamic load scheduling. The framework is based on automated and dynamic load scheduling with minimal assumptions about the future events and partial information sharing to protect consumer privacy.

A μG has different types of consumers with different types of electrical loads. Each of these load types have their characteristic properties. These demand can be met by the electricity supply from different energy sources including renewables. The renewables provide a cheap and environmental friendly alternative to expensive electricity from fuel-based power plants. However, the time varying and intermittent nature of the renewables makes the task of integrating them into the electrical grid challenging. Also, demand from the consumers are not deterministic and changes with time. The focus of this research is to dynamically schedule loads (electrical demands) from the consumers at optimal time to achieve energy efficiency in the μG measured in terms of the total electricity cost of the consumers. The models also investigate the design problem of storage sizing through trade-offs between different storage sizes and total electricity cost.

The major contributions of this research include:

- Developing a framework for integrating building loads with production line loads from different types of consumers
- Developing a dynamic load scheduling model for automated online load scheduling with partial information sharing
- Developing a generalized framework for online sequential load scheduling decisions using reinforcement learning

1.6 Organization of the thesis

The remaining of this Thesis is organized as per the following sequence. In Chapter 2, literature relevant to each of the research topics is reviewed to identify gaps in the literature and use existing literature to build on this research. Chapter 3 presents mathematical models for automated load scheduling and storage sizing with a single process parallel machine shop. The models in Chapter 3 are based on complete information sharing to develop a foundation for Chapter 4. Chapter 4 considers dynamic scheduling in a complex job shop environment while considering on site energy harvest and future estimations. Chapter 5 presents a dynamic scheduling model for dynamic load scheduling using partial information sharing. In Chapter 6, a framework for reinforcement learning is presented that could be generalized to different online problem environment under certain conditions.

Table 1.1.
Organization of the thesis

Chapter	Title
Chapter 1	Introduction
Chapter 2	Literature Review
Chapter 3	Storage Sizing and Optimal load scheduling
	- Static approach using mixed integer linear programming
	- Dynamic approach using Lyapunov optimization
Chapter 4	Intelligent algorithm for dynamic job shop scheduling
Chapter 5	Dynamic load scheduling with Partial Information Sharing
Chapter 6	Reinforcement learning for end-to-end dynamic load scheduling

2. LITERATURE REVIEW

Electricity supply chain is a vast area of research and involves a lot of domains e.g. engineering, management, economics, government, utility companies, policy makers and, consumers. The major focus of this research is to develop framework and models to achieve energy efficiency in electrical grid through operational practices. In this chapter, literature review is provided for the approaches, problems and solution methodology adopted in the literature.

Energy efficiency can be achieved by using energy efficient devices, newer machines, transmission and distribution lines with lesser energy losses, high capacity storage and, highly efficient power plants. However, these methods require huge investment and could not accommodate the existing infrastructure. Another method is using the available energy efficiently through operational practices such as changing the demand pattern of consumers, increasing the utilization of the electricity generators and increasing the reliability of the grid by using distributed electricity resources. Different problems have been solved as part of achieving energy efficiency in the grid. The definitions of these problems with some previous works are provided as follows:

Definition 5. Demand response (DR)

DR refers to the mechanisms that the utilities use to encourage consumers to curtail or shift their load in order to reduce aggregate demand at particular times [33]. DR is poised to play an important role in the smart grid [34]. DR can be price driven or event driven ([35], [36]). DR programs have limits on the number of events it can trigger [36]. Chen et. al [36] state that DR programs could be instrumental in reducing peak demand but to achieve the desired goal, DR programs must be used judiciously, i.e., when peak loads are very high and by spreading the DR participation fairly across all customers [37].

Definition 6. Demand side management (DSM)
DSM refers to the load control actions taken by the consumers in response to the DR strategies of the utilities. DR and DSM are often used interchangeably. DSM is from the consumer's side while DR is set from the utilities side of the electrical grid. Demand task scheduling based on the controlled release threshold policy developed in Koutsopoulos et. al [38] aims at minimizing the grid operational cost over the time horizon. Imamur er.al [39] proposed a genetic algorithm approach to match the load profile with optimal coast based load profile. Vasirani et.al [40] proposed joint demand profile model of smart consumer load balancing, where consumers actively participate in the balancing of demand with supply by forming groups to be contracted in the market through an aggregator.

Definition 7. Unit commitment (UC)
UC problem is defined as an optimization problem to solve generation schedule of electricity generating plants. The decisions include which plants to be turned on, when and run at what levels. UC problem considers non-shiftable loads with known demand distribution. Carpetier [41] adopted a stochastic approach to solve UC for generating optimal supply curves. [42] Logenthiran et. al studied UC for microgrids working in an islanded state. The problem also includes selection of renewable power plants [43]. Review of UC problems and different solution approaches is provided in Saravanan et. al [44]. Review for stochastic optimization for UC is discussed in Zheng et. al [45].

The focus of this research is to develop models to enable us to achieve energy efficiency in the μGs through automated load scheduling. Automated load scheduling integrates the different problems as:

- SO sends messages $m_i[t]$ to DLCs and load are scheduled by DLCs based on these messages. Thus, SO sends *Demand Response* signals which is followed by the consumers through their respective DLCs. This response to SO messages is the *Demand Side Management* by consumers.

- Integrating load scheduling with supply side aims at obtaining the optimal schedules for running fuel based electricity generators. This is the integration of *Unit Commitment* in μG.

2.1 Electrical grid

The electrical grid is an interconnected network for delivering electricity from suppliers to costumers [46]. The current electric grid is aging and needs significant changes to meet the growing demand. Demand for electricity is growing at a much faster rate than what can be met through expansion of the conventional grid. Uncertainty in demand and generation and volatility by penetration of renewable resources bring great challenges in grid management [43]. Since the electricity cannot be stored because of very small storage capacity, fluctuations in demand can be met by over production which makes the grid inefficient. Grid generates electricity to meet peak demand and rest of the capacity is underutilized most of the times [47]. It shows that current electrical needs a revamp and major structural and operational changes.

One of the structural changes include converting the electrical grid into an interconnected network of μGs as discussed in Section 1.2. μGs will gradually be a strong and effective support for the main power grid and potentially one of the future trends of power system [48]. The future electrical grid is envisioned to be a network of interconnected μGs([8], [3], [7], [49], [50]). One of the operational changes include active participation of consumers in load scheduling as discussed in Section 1.3. The focus of this research is to develop a framework to solve automated load scheduling problem in a μG.

2.2 Energy efficient scheduling (EES)

This research aims to integrate different types of consumers in a μG in load scheduling. Industrial consumers have production line loads that are different from other types of load discussed in Section 1.2.2.

2.2.1 EES in residential houses and commercial buildings

Number of smart homes with smart devices capable of intelligent load management are increasing rapidly as shown in Figure 1.16. Some of the works consider a single house or commercial building while some of the works include a cluster of house and commercial buildings in a neighborhood. Thus, the load scheduling for residential houses and commercial buildings has been studied but independent of industrial consumers. Mohsenian et. al([19]) developed load scheduling model for residential loads under real-time pricing policy. Amir et. al [24] proposed autonomous demand side energy management in a static system among residential consumers. Fuller [34] provided an analysis for price based DR in residential consumers. Vasirani et. al ([18]) proposed an optimization model for demand supply balance based on aggregating load profile of consumers. Bakr and Cranefield [51] considered optimal load scheduling in a residential neighborhood through demand aggregation. Energy consumption of certain appliances in a household is studied by Baharlouei et. al [52]. Wu et. al [30] developed an approximation algorithm for optimal load scheduling among residential consumers. Song et. al [21] developed non-stationary load scheduling strategies using game theoretic approach for residential consumers. Huang et. al ([28]) developed optimization model for minimizing total electricity cost by sharing among residential consumers. Bhosale et.al([20]) proposed a framework for consumption scheduling among residential consumers. Hu et al. [25] considered load shifting for minimizing total electricity cost in a residential house. Home energy management aimed at maintaining comfort levels is shown studied in Liu et. al [53].

2.2.2 EES in industries

Energy efficient scheduling has attracted researchers and is being studied extensively. Gahm et. al [54] and Giret et. al [55] presented a nice literature survey on energy efficient load scheduling in industries. The industrial loads are significantly different from residential and commercial building loads [17]. Magnitude and ability to control their demand makes industries good candidate for inclusion in DSM and DR practices [56]. Starke et. al [57] explains why industries are more ready for DSM than residential an commercial sector. May et. al [58] develops 7 KPIs for energy efficiency in production systems. Role of information systems in energy management for efficiency is discussed in Zampou et. al [59]. Energy management in industry 4.0 scenario is discussed by Shrouf et. al [60].

Figure 2.1 shows different levels at which the energy efficient optimization problems are formulated for industries. In Level 1, DR strategies are formulated by the utilities as part of the strategic planning. At Level 2, industries formulate optimization problems to achieve their individual objectives as part of process planning. At Level 3, industries formulate optimization problems for a particular process in process planning as part of operational planning ([61], [62], [63], [64], [65]). The focus of this research is on integrating strategic planning with process planning such that overall efficiency in a μG can be achieved to enable energy efficiency in the electrical grid.

Different architectures and conceptual designs for sustainable manufacturing have been presented by researchers. Trentesaux et.al [66] proposed green Holons for manufacturing systems. Thomas et.al [67] leads a discussion on using intelligent manufacturing systems and services for sustainability. [68] provided a quantitative report on potential energy savings from multi-robot assembly systems. Different approaches and trends in sustainable manufacturing is discussed in Trentesaux and Prabhu [69].

In this section, a brief taxonomy is presented in Figure 2.2 that shows gaps in the literature. Each cell in Figure 2.2 is represented by $W(w)$ where W is the cell reference and w is the number of research articles found according to the taxonomy. The reference for the research articles is shown in Table 2.1. It can be seen from Figure 2.2 that the research on job shop scheduling, particularly dynamic scheduling is scarce. Static scheduling refers to the schedules that are fixed and not changed over next time periods. Reactive scheduling refers to the schedules that are changed or revised based on the current state or future expected state. Dynamic scheduling refers to developing load schedules based on the current information. Exact approach considers

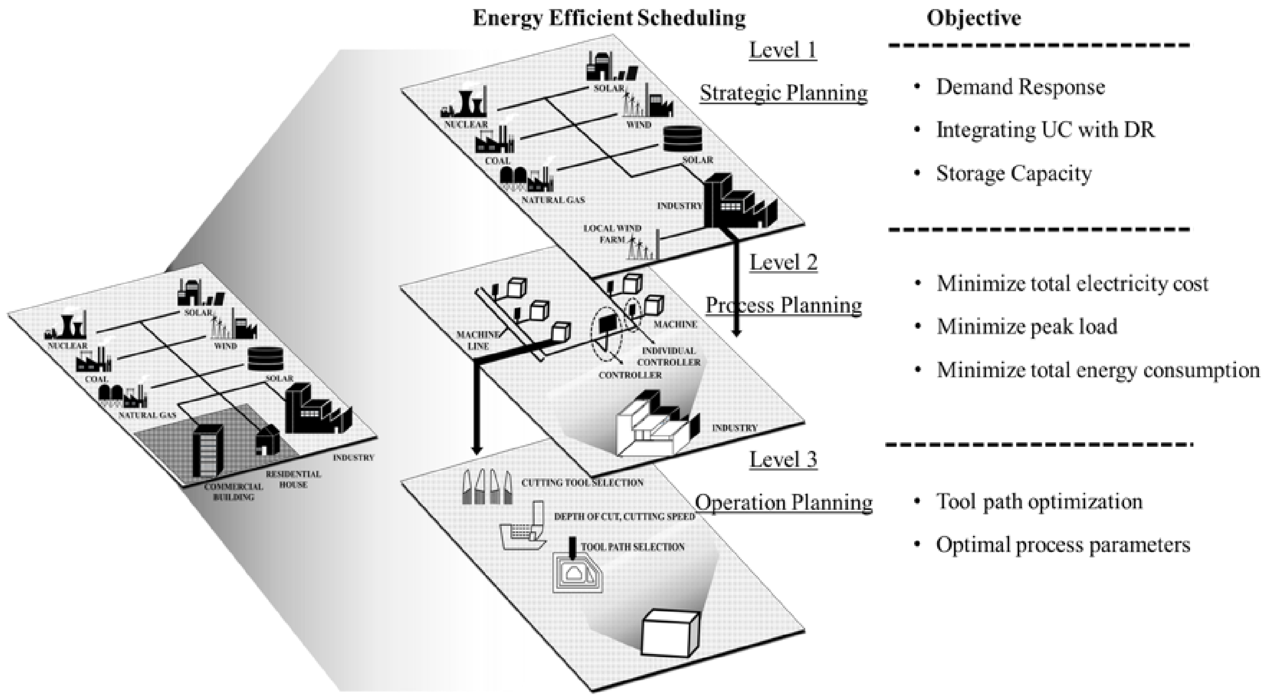


Fig. 2.1. Schematic representation of a μG

solution approaches that guarantee an optimal solution. Heuristics approach considers problem specific heuristics and meta-heuristics to solve the problem. Simulation approach refers to a simulation model built in the paper. Some paper considered analytic approaches to analyze the results but did not consider finding the optimal value of the objective function.

Static manufacturing Scheduling in Smart Grid Scenario

As it can be seen from Figure 2.2, most of the existing works in manufacturing scheduling are in static scheduling models for flow shop problems. Nayak et. al ([70]) discuss the use of a static scheduling approach using Genetic Algorithms for multi-objective flow shop scheduling problem. The article showed that the scheduling problems can be solved only for the simplest case of manufacturing scheduling. Even

		Industrial Consumers				
		Single process	Continuous	Flow shop	Job shop	Assembly
Energy perspectives	Peak Demand	1 (0)	14 (2)	27 (7)	40 (0)	53 (0)
	Total electricity consumption	2 (15)	15 (7)	28 (29)	41 (15)	54 (3)
	Total electricity cost	3 (18)	16 (10)	29 (17)	42 (7)	55 (2)
	Carbon footprint	4 (2)	17 (0)	30 (8)	43 (4)	56 (0)
	Electricity as constraint	5 (6)	18 (1)	31 (5)	44 (1)	57 (0)
Approach	Exact solution	6 (16)	19 (14)	32 (16)	45 (5)	58 (1)
	Heuristics	7 (20)	20 (4)	33 (16)	46 (20)	59 (1)
	Simulation	8 (3)	21 (4)	34 (0)	47 (1)	60 (1)
	Analytical	9 (2)	22 (0)	35 (1)	48 (1)	61 (2)
Scheduling type	Reactive	10 (3)	23 (1)	36 (5)	49 (4)	62 (0)
	Static	11 (31)	24 (16)	37 (25)	50 (25)	63 (2)
	dynamic	12 (3)	25 (2)	38 (2)	51 (2)	64 (1)

Fig. 2.2. Taxonomy for industrial energy efficient scheduling

for the simplest case, it could not be solved for more than 2 machines. Evolutionary computing was employed to solve the multi-objective problem. However, the static scheduling models based on evolutionary algorithms might not be useful in a smart grid scenario due to the following reasons:

- static scheduling models either neglect the real time fluctuations in stochastic variables e.g. electricity price, consumer demand, energy from renewables or they assume perfect information from stochastic variables which is impractical
- static scheduling models do not consider real-time information. When renewable sources of energy are considered, this might lead to significantly poor solutions [71].
- static scheduling models could be used in multi-stage scheduling e.g. taking recourse action. However, since it is already difficult to solve the scheduling

problem, taking recourse action might not be feasible in dynamic scheduling based on real - time information.

These shortcomings of static scheduling models with respect to the smart grid has recently drawn the attention of the researchers in dynamic scheduling models. A large proportion of existing works on dynamic scheduling models is in the field of communication networks. In these problems, there are no external stochastic variables as against smart grid. These stochastic variables in smart grid makes the problem more complex.

This Section is not intended to belittle the evolutionary computing methods. It is an observation that evolutionary computing methods may not be applicable in smart grid scenario in its current form. Even though the static scheduling problems may not be useful for smart grid scenario, the existing standard evolutionary methods can be modified to suite the needs of the dynamic nature of the smart grids as explained in the future directions in Section 6.8.

Dynamic manufacturing Scheduling in Smart Grid Scenario

As argued in the previous section, dynamic scheduling models are more useful for smart grid scenario due to its ability to capture real-time information and handle the uncertainty in stochastic variables. In this research, the focus is on dynamic scheduling in job shop manufacturing facility. Hao et. al [72] discussed a flow shop problem in smart grid scenario, however, the Job shop manufacturing scheduling has not received much attention in literature. Dynamic scheduling has computational advantage as the the problem need not be pre-solved. Also, dynamic scheduling can handle online demand (customer requests) which static approach assumes full information of future demands. Based on the gaps as shown in through the Table 2.2, this research discusses a dynamic load scheduling in job shop manufacturing environment from consumer's perspective in Chapter 4.

Table 2.1.: Research papers in Figure 2.2

Cell	Research papers	Cell	Research papers
1	-	2	[73], [74], [75], [76], [77], [78], [79] , [80], [81], [82], [83], [84], [85], [86], [87]
3	[88], [89], [90], [91], [92], [93], [94], [95], [96], [97], [98], [99], [100] , [101], [83], [84], [102], [103]	4	[104], [82]
5	[105], [106], [91], [107], [108], [109]	6	[73], [105], [106], [88], [90], [92], [93], [95], [98], [80], [100] , [83], [84], [104], [85], [103]
7	[74], [75], [78], [79], [89], [91], [107], [92], [94], [96], [97], [99], [81], [82], [101], [82], [102], [108], [86], [87]	8	[76] [77], [96]
9	[101], [109]	10	[89], [94], [101]
11	[73], [105], [106], [78], [79], [88], [90], [91], [107], [92], [93], [95], [96], [97], [98], [80], [99], [100], [81], [82], [110], [83], [84], [104], [82], [102], [108], [109], [85], [103]	12	[74], [75], [86]
13	-	14	[111], [112]
15	[113], [114], [115], [116], [110], [110], [117]	16	[118], [119], [120], [121], [91], [122], [123], [124], [125] [126]
17	-	18	[127]

continued on next page

Table 2.1.: *continued*

Cell	Research papers	Cell	Research papers
19	[118], [119], [120], [121], [113], [114], [115], [127], [123], [116], [110], [110], [117], [125]	20	[122], [115], [116], [124]
21	[111], [128], [91], [126]	22	-
23	[127]	24	[118], [119], [120], [111], [128], [121], [113], [114], [115], [127], [123], [116], [124], [110], [117], [125]
25	[91], [122]	26	-
27	[129], [130], [131], [132], [133], [126], [134]	28	[135], [135], [136], [137], [138], [139], [140], [141]
29	[129], [142], [143], [144], [145], [146], [133], [147], [148], [149], [150], [103], [140], [151], [152], [134], [153], [154]	30	[135], [135], [130], [142], [132], [138], [155] , [156], [87]
31	[157], [131], [158], [159], [157]	32	[129], [130], [142], [131], [158], [137], [159], [145], [133], [147], [148], [103], [140], [151], [140], [153]
33	[157], [136], [138], [143], [144], [139], [155] , [157], [156], [149], [134], [150], [152], [134], [141], [154]	34	-
35	[146]	36	[157], [156], [133], [147], [148]

continued on next page

Table 2.1.: *continued*

Cell	Research papers	Cell	Research papers
37	[157], [135], [135], [129], [130], [142], [131], [137], [138], [159], [143], [144], [139], [145], [146], [155], [134], [150], [103], [140], [151], [152], [134], [140], [141], [153], [154]	38	[158], [149]
39	-	40	-
41	[160], [161], [162], [163], [164], [165], [166], [167], [168], [169], [170], [171], [172], [173], [168]	42	[174], [156], [175], [176], [177], [178], [179]
43	[180], [181]	44	[160], [182], [183]
45	[160], [176], [165], [166], [167]	46	[174], [156], [162], [180], [181], [175], [177], [163], [169], [164], [167], [168], [170], [171], [172], [173], [178], [168], [182], [183]
47	[161]	48	[179]
49	[132], [161], [174], [169]	50	[136], [132], [160], [174], [156], [162], [177], [163], [164], [165], [166], [180], [181], [175], [176] [167], [168], [170], [182], [171], [172], [173], [178], [168], [182], [183]
51	[184], [179]	52	-
53	-	54	[185], [186], [187]
55	[185], [188]	56	-

continued on next page

Table 2.1.: *continued*

Cell	Research papers	Cell	Research papers
57	-	58	[185]
59	[187]	60	[186]
61	[188], [186]	62	-
63	[185], [187]	64	[188]

Few works consider integrating supply side with demand side. Logenthiran et. al [42] studied short term generation scheduling in μGs. Central and local energy management is studied in [22]. Khodaei et. al [189] developed a co-optimization model for μG investment planning on generation and transmission lines. [190] studied trade-offs between storage capacity and total electricity cost for storage sizing when multiple μGs interact. [191] studied an integrated model for storage sizing and energy management in a μG.

Literature on integrating all the consumers, particularly industrial consumers with different production line loads, is scarce. Zhang et. al [192] considered integrating industrial load scheduling with residential and commercial consumers for cost minimization of industrial consumers given information on the energy usage from other consumers. Fin et. al [193] studied electricity consumption pattern between two industries under real time pricing. Young et. al [194] presented an analysis into energy consumption in production line along with manufacturing robots. Zhang et. al [195] developed real-time pricing model for collaborative industries. However, the work did not consider residential or commercial consumers. Sun et. al [196] considered integration of manufacturing loads with HVAC systems.

This research identifies two major gaps as listed below and focuses on them to bridge the gap in the literature in the capacity of μGs.

- Dynamic scheduling for flexible job-shop has not been discussed in the literature as pointed out in Stock et. al [197]

- Most of the work does not integrate different types of consumers in a smart grid scenario and solve for different consumer types independently

- most of the existing works consider either a consumer's perspective or a grid operator's perspective. This research focuses on integrating the two perspectives.

Adoption of μGs is challenging and requires an interdisciplinary effort from engineering, economists, academia, industries and government. Kema technologies [7] lists the different challenges to be faced before bringing MGs to mainstream. Due to the different sources of generation, μGs face technical difficulties in voltage and frequency regulation [9]. Challenges also include public response and government policies. μGs face the problem of: *"Not in my backyard"*. Despite challenges, μG promises a potential for green electrical grid.

2.3 Lyapunov Optimization

Lyapunov Optimization, also known as stochastic dual gradient, is a policy aimed at minimizing cost functions for sequential time slots that are additive. The policy aims at minimizing the average of total accumulated cost while achieving mean rate stability of the queues, where the queues may be real or virtual. It has been used for different problems with finite and infinite period. Lyapunov optimization is has been adopted in different applications ([198], [199], [200]) and we refer the readers to textbook by Neely ([201]) for a tutorial on Lyapunov optimization. [202] explains Lyapunov optimization as a stochastic dual gradient method. Lyapunov optimization has been extensively used for DSM in smart grid because of its ease of implementation and performance guarantees. Neely ([203]) proposed online algorithm for energy allocation. A distributed optimization approach was investigated in [29]. Salinas et. al ([204]) considered dynamic scheduling with delay tolerant and intolerant loads. Quality of service for residential consumers considering luxury usage is discussed in [205]. Lakshminarayana et. al ([206]) used Lyapunov Optimization to study the interaction between different μGs and their storages. Qin et. al ([50]) implemented Lyapunov optimization in storage networks. Li et.cal ([207]) studied finite time horizon problem. Shi et. al ([208]) considered Lyapunov optimization from the perspective of a central controller when the demand from the consumers is fed into the central controller. In this research, a new policy is developed based on Lyapunov optimization where:

- Future expectations are considered while making decisions at time slot t. A recurrent neural network and time series model is constructed that forecasts energy available from the renewables in the near future time slots
- A multi-objective Lyapunov optimization is considered where two-objectives are converted to time averaged constraints

2.4 Game Theoretic approach for load scheduling

As the problem of load scheduling involves different consumers who want to minimize their electricity cost and schedule their loads when required (to maximize their utility) under the supervision of a central manager (SO), the problem sets up for a game theoretic approach with N players and 1 manager. Different game theoretic approaches have been adopted in the literature. Amir et. al [24] formulated an energy consumption scheduling game by proposing a static strategy and unique Nash equilibrium. Bakr and Cranefield [51] proposed a static fair billing mechanism to ensure fairness in billing among different consumers based on their consumption. Vasirani and Ossowski [18] developed a collaborative model among cooperating consumers for load management. Wu et. al [30] proposed a static day-ahead strategy for load scheduling and solved the problem using polyblock approximation algorithm. A dynamic Stackelberg game model for residential load scheduling for a single consumer is presented by Yu and Hong [209].

However most of the approaches consider either static strategy or complete information sharing that poses a privacy and security concern. In this research, a dynamic scheduling framework is used with partial information sharing such that the consumer privacy and security are not compromised. Neely [200] proposed repeated stochastic game using Lyapunov optimization where control decisions are taken by the manager for load scheduling among N participants.

2.5 Demand and resource sharing

Demand and supply in the electricity market is highly unpredictable with a lot of uncertainty. In this research, where the electrical grid is envisioned to be an interconnected network of the μGs, collaboration between interested μGs could be used to mitigate time varying and intermittent nature of electricity supply and demand. Demand and capacity sharing can be used to maximize profit and resource utilization, enable timely delivery to customers in spite of the uncertain market demands and unexpected capacity shortages, and maximize the overall stability of the system [210]. The μG may collaborate by sharing demand and capacity when required based on mutual contracts. Demand and capacity sharing has not been studied from the context of electrical grid. It has mostly been studied for internal collaboration within an organization or vertical collaboration between organizations [211].

Collaboration among companies could also enhance the sustainability of the system in the long term. For effective and fair sharing among the participants, protocols play a significant role. Protocols are decision rules or or procedure to be followed when two entities collaborate. Yoon and Nof [212] have implemented the decision protocols for effective demand and capacity sharing among enterprises: demand sharing protocol and capacity sharing protocol. These protocols focused on real-time information sharing without considering long term analysis. Seok and Nof [213] presented demand and sharing protocols with long term analysis of collaboration among organizations.

These collaboration among different entities can either be static or dynamic, that is, sharing relationship between entities may change over time. Yoon and Nof [214] studied affiliation and dissociation among collaborating organizations when collaboration rules or conditions change. Adaptive demand and capacity sharing is studied in Seok and Nof [211]. However, the collaboration incurs cost because of the contracts between the organizations and cost of protocol invocation every time a protocol is invoked for sharing. Moghaddam and Nof [210] developed protocols for dynamic best matching among collaborating organizations. They also provide a mechanism for real-time optimization in collaborative network environment [215]. Moghaddam and Nof [216] provide an extensive explanation of best matching protocols among collaborative organizations or entities.

When multiple entities collaborate to achieve a common goal, fairness must be ensured when building the collaborative protocols. Social welfare function has been extensively used in economics as a measure of income inequality and comparing income distributions. It has been used in different applications to enforce fairness in decision making with regard to different allocations. Social welfare function has been used in resource allocation for wireless network ([217], [218], [219]), ambulance dispatch [220], task allocation in multi-robot system ([221], [222]) and, decision making in intelligent shared environments [223]. Nayak et al. [224] studied social welfare function in smart grid scenario based on resource and task sharing. Fairness in this research is considered in terms of balancing the delays in scheduling of the requested loads.

In this research, a simple notion of centrally controlled sharing among two μGs is considered where the two μGs collaborate to share demand and capacity by the supply of electricity when required from their renewable resources. They share demand as demand of one μG could be met by the supply of electricity from the other μG. They share capacity as battery and sources of energy from one μG can be used to serve the demand from the other μG. This sharing makes intuitive sense as energy can be stored in different forms e.g. heat energy, kinetic energy or potential energy. If the location of one μG is suitable for energy storage, it can store more energy if available from the other μG. Similarly, demand can also be shared indirectly through supply of electricity from one μG to another μG.

2.6 Reinforcement learning for load scheduling

Reinforcement learning is an area of artificial intelligence (AI) where the agents learn state value functions, state-action value functions and policies to take actions in an environment to maximize its reward [32]. While machine learning has achieved significant success in recent time, particularly using deep learning, machine learning is a subset of Artificial Intelligence, like Reinforcement Learning. The relationship between reinforcement learning and machine learning is shown in Figure 2.3. The

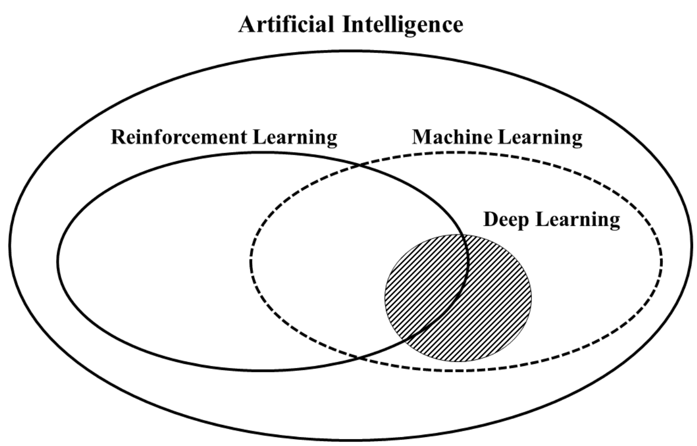


Fig. 2.3. Schematic representation of Artificial Intelligence (AI)

behavior of the environment, the rewards and the system dynamics are generally not known in advance and the agents have to learn them to make optimal decisions such that their cumulative reward is maximized [225]. The iterative procedure of learning by agents is shown in Figure 1.18 where the agents interact with the environment to understand the environment function, system dynamics, rewards from different actions and the value of being in different states [32].

Value function iteration and policy iteration have been commonly used in the past to learn the state and value functions for energy efficiency. Q- learning is value iteration method where values of the state-action pair are obtained. Different methods to perform reinforcement learning based on value function iteration and policy iteration is discussed and explained in the text book by Sutton and Barko [32]. Aydin and Ostemel [226] used improved Q-learning for scheduling different jobs by setting job priority in a job shop environment. Glaubius et.al [227] developed a value iteration method for dynamic resource scheduling. Kara et.al [228] used Q-learning for scheduling smart devices in a residential building in a smart grid. Li and jayaweera [229] developed a Q-learning model for dynamic load scheduling in a residential set-up in smart grid. Kim et. al [230] used Q-learning for dynamic electricity pricing. Zhang and Schaar [231] developed an improved value iteration method for load scheduling in smart grid. Qing et. al [232] used Q-learning for optimal electricity generation control in the electric grid. Yu et.al [233] used Q-learning for collaboration amongst different generators in the smart grid. Kuate et. al [234] used reinforcement learning for pricing of the electricity based on the current state. Wang et. al [235] used Q-learning for residential load scheduling with on site energy generation and energy storage facility. Rayati et. al [236] developed a policy iteration method for energy efficient load scheduling with residential buildings. Pan [237] used Q-learning for lighting scheduling in a residential building. Jay and Swarut [238] used policy iteration method for unit commitment problem. Ahmed and Bouffard [239] developed a single Markov Decision Process (MDP) for clustering different consumers for efficient load management. Ruelens et. al [240] used reinforcement learning for optimal thermostat control in residential building. Chen et. al [241] studied a single machine single operation machine scheduling problem and built a deep learning based model. Asl el. al [242] developed a framework for multi-objective reinforcement learning based on value iteration to get Pareto optimal solutions.

However, value based methods work only when the system dynamics is well defined or when the action and state space is small. Dynamic load scheduling problem has high-dimension and continuous state and action space. Thus, it is not practical to use traditional value iteration and policy iteration methods. Approximate function estimation can be used to learn functions that map from the state to actions or state to state values or state to state-action values. In this research, a deterministic policy is learnt that maps from state values to action values. There are some works on approximating function values for load scheduling in smart grid. Jasmin et. al [243] used a function approximation for approximating state -action values for solving a unit commitment problem. Zhang et.al [244] developed a deep Q-learning based reinforcement learning method for supply and demand oriented Stackelberg game in smart grid. The deep artificial neural network model acts as critic for the value iterations. Bahrami et.al [245] developed an online learning algorithm for demand response in a smart grid scenario. They considered different load types and service quality through delays and built an actor- critic method for learning optimal actions. They also used value function approximation instead of value iteration to obtain state function values. Policy iteration method is much more stable than the value iteration methods.

Policy search performs better as compared to value iteration methods because of its stability. Also, as the dimensionality and cardinality of state and action space increases, value function methods are not practical. But even though policy search is more stable, as the number of parameters needed to approximate the policy function increases, the number of steps in Figure 1.18 for convergence increases exponentially. When neural networks in used for approximating the policy, any function can be approximated but the number of parameters is very high. In dynamic load scheduling, this research approximates a function from current state to the load scheduling actions. To handle the increased number of steps requirement, a guided policy search method is employed.

In guided policy search, the policy search is converted into a supervised learning problem. Supervised learning is more robust and stable than policy search. The policy is optimized using supervised learning, which scales nicely with the number of dimensions of the action and state space. The training set for supervised learning can be constructed using trajectory optimization under known dynamics [246]. Another issue with most of the practical applications is that the neither system dynamics nor the systems states are actually known. Thus, in this research, a deterministic policy is developed using the guided policy search that provides actions based on the observations provided by the system. One major issue with policy search is that is very slow. It is slow as it needs two steps: convergence and improvement. The the problems are difficult in itself. However, the two problems can be solved simultaneous using guided policy search as discussed by Levine et.al [225].

In guided policy search, locally optimal trajectories are obtained and are used as response variables for the training part of the supervised learning. Since the locally optimal actions are known, it is used to generate the system dynamics from observations. Thus, both the target and the behavior policies are the different. Guided policy search converts policy search into supervised learning, by iteratively constructing the training data using an efficient model-free trajectory optimization procedure [225]. In this research, the guided policy search is used to obtain deterministic policy. A deterministic policy maps system observations/ states to actions [247]. The idea of using neural network for approximating the deterministic policy is taken from Lillicrap et.al [248]. The optimal trajectories in this research are based on sample based paths used by Lioutikov et.al [249]. A distribution is obtained for optimal actions rather than just a single action value for achieve robustness and create smoothness in the functions as non-smoothness makes the learning process slow [246]. In the guided policy search, the deterministic policy is trained such that the guided trajectory distribution converges to the learnt parametric policy. Even though, locally optimal trajectories are used to generate data for the supervised learning, the convergence in policies ensures that the policy produces a good long-horizon performance. In this research, the optimal trajectories are obtained by solving the load scheduling problem at the end of the week. Thus, every week has as different starting state. This helps in making the policy more robust to different starting states and making the best decisions without making compounding errors.

Guided policy search has drawn the interest of researchers because of several advantages it brings to reinforcement learning. The major advantages include:

- it converts reinforcement learning to supervised learning problem that is more stable and thus converges faster
- the locally optimal trajectories reduce the number of data points required for learning the deterministic policy
- in load scheduling problem, individual weeks can be assumed to be fairly independent and since a deterministic problem can be solved at the end of the week to optimality, response variables in the supervised learning are optimal actions, thus reducing the need for policy improvement step.

2.7 Data and Softwares used in the research

To develop realistic case studies, real data have been used in this work. The electricity prices for $p_2[t]$ are taken from summer day price by the Ameren power company [250]. Non-shiftable load profile for different consumer types is obtained from the DOE website [251]. Data for average hourly electricity harvest from renewables is obtained from the ERCOT website [252]. The data for factory scheduling (production line loads and processing times) have been simulated as discussed in the research article by Fang et. al [98] and the electrical loads for residential and commercial buildings have been taken from the research article by Hao et.al [72].

This research is conducted on a Windows i7 processor on a 8GB RAM laptop. The codes were written in Python and all the codes are available at Github [253]. Eclipse and PyCharm IDE were used to write the codes([254], [255]). For optimization, GUROBI [256] Student license was used and the computational times mentioned in this research are the time (in seconds) when the solver finds the optimal solutions within its tolerance limits. Tensorflow by Google [257] is used to develop the prediction function and a 2 layer recurrent neural network architecture is selected for predicting wind energy from the renewables. A three layer perceptron architecture is considered to model the deterministic policy in reinforcement learning.

3. STORAGE SIZING AND OPTIMAL LOAD SCHEDULING

Most of the load scheduling problems solve the problem independently for non-industrial and industrial consumers. In a μG where different types of consumers share electricity generation resources and storage, the consumers are coupled as consumption pattern of one consumer affects the electricity cost of other consumers. Since storage are expensive, optimal storage sizing is critical for μG design. In this chapter, we investigate optimal storage sizing under two different scenarios 1) long term load scheduling 2) dynamic load scheduling. In both these scenarios, we consider residential houses, commercial buildings and a manufacturing facility as consumers who participate in automated load scheduling. The objectives of this chapter is to:

- Construct mathematical models to integrate production line load scheduling with other types of load and consumers in a μG.
- Analyze electrical storage capacity under different objectives using trade-offs between the total electricity cost and storage capacity.

The mathematical models have been constructed based on certain assumptions as mentioned in previous chapters. The summary of the assumptions are as follows:

- The μG considered is a direct current μG and voltage synchronization has been taken care of by converting the alternating current electricity into direct current.
- The consumers participate in automated load scheduling and the smart devices have the capability to respond to the real-time load scheduling decisions.
- The electrical loads of different devices are deterministic and known.
- The time horizon is one day that is divided into different discrete time slots.

- Discrete time model is considered because real-time prices are released in discrete time at an interval of 15 minutes.

- The loads take integer multiple of time slots to complete. That is, all the loads run for integer multiple of 15 minutes.

- There are no outage costs. If the demand can be met from μG, electricity is bought from Macrogrid and the capacity of Macrogrid is ∞.

- Set-up time in the production line loads are included in the processing time.

To construct the mathematical models, the following set of constraints for storage are used in every model. Equation 3.1 shows the evolution of electrical energy in the battery where η is the storage efficiency. Equation 3.2 ensures feasible discharge from battery. Equation 3.3−Equation 3.6 state the resource constraints. Equation 3.8 is the variable type constraint. Since the decisions are taken at time t, the scheduling decisions will always draw available energy from the battery. This frequent charging and discharging of the battery affects the life of the battery. To handle the problem of frequent charging and discharging from the battery, a cost cd is used in the objective function and a binary variable $\bar{c}[t]$ is used if energy is drawn from the battery as shown in Equation 3.7 where M_B is a big number.

$$A[t+1] = \eta A[t] + Y[t] - B[t] \qquad \forall t \in \{1, 2, ..., T\} \tag{3.1}$$

$$B[t] \leq A[t] \qquad \forall t \in \{1, 2, ..., T\} \tag{3.2}$$

$$B[t] \leq B_{max} \qquad \forall t \in \{1, 2, ..., T\} \tag{3.3}$$

$$Y[t] \leq Y_{max} \qquad \forall t \in \{1, 2, ..., T\} \tag{3.4}$$

$$A[t] \leq A_{max} \qquad \forall t \in \{1, 2, ..., T\} \tag{3.5}$$

$$B_{max}, Y_{max} \leq A_{max} \qquad \forall t \in \{1, 2, ..., T\} \tag{3.6}$$

$$\big(B[t] + Y[t]\big) \leq \bar{c}[t] M_B \qquad \forall t \in \{1, 2, ..., T\} \tag{3.7}$$

$$A[t], B[t], Y[t] \in R^+ \qquad \forall t \in \{1, 2, ..., T\} \tag{3.8}$$

The storage options considered in this chapter are shown in Table 3.1. The different storage sizes are taken from the paper by Lakshminarayan et. al ([206]). The analysis on different storage sizes is presented in this Chapter without considering the investment cost for the storages since the focus is on short term scheduling.

Table 3.1.
Different Storage options

Option	A_{max}	B_{max}	Y_{max}
1	0.8 MW	0.2 MW	0.2 MW
2	2 MW	0.4 MW	0.4 MW
3	4 MW	0.8 MW	2 MW
5	20 MW	4 MW	4 MW

The μG has residential houses and commercial buildings as non-industrial consumers and a parallel machine shop as industrial consumer. The consumers may have different load types and each load type has its own characteristic constraints which have to be satisfied. The constraint sets have been constructed in this section to be used in different models as they remain valid for all the models throughout the thesis.

Constraint set for the time shiftable loads is constructed in Equation 3.9 − Equation 3.11. Time constraint is stated in Equation 3.9. Continuity of the time shiftable load at constant power is ensured by Equation 3.10. Variable types is shown in Equation 3.11.

$$\sum_{t=t_{i,r}^{ts,min}}^{t_{i,r}^{ts,max}} x_{i,r}^{ts}[t] = 1 \quad \forall r \in R, \forall i \in I \tag{3.9}$$

$$x_{i,r}^{ts}[t] \leq x_{i,r}^{ts}[t^{'}] \quad \forall t \leq t^{'} \leq t + tr_{i,r} \tag{3.10}$$

$$x_{i,r}^{ts}[t] \in \{0, 1\} \tag{3.11}$$

Constraint set for the modifiable power loads is constructed in Equation 3.12 − Equation 3.15. Energy requirement constraint for modifiable power loads is shown in Equation 3.12 while power demand constraints are shown in Equation 3.13 and Equation 3.14. Variable type constraint is shown in Equation 3.15.

$$\sum_{t=t_{i,h}^{ps,min}}^{t_{i,h}^{ps,max}} x_{i,h}^{ps}[t] = L_{i,h}^{ps} \quad \forall h \in H, \forall i \in I \tag{3.12}$$

$$x_{i,h}^{ps}[t] \leq L_{i,h}^{ps,max} \quad \forall h \in H, \forall i \in I, \forall t \in \{t_{i,h}^{ps,min}, ..., t_{i,h}^{ps,max}\} \tag{3.13}$$

$$x_{i,h}^{ps}[t] \geq L_{i,h}^{ps,min} \quad \forall h \in H, \forall i \in I, \forall t \in \{t_{i,h}^{ps,min}, ..., t_{i,h}^{ps,max}\} \tag{3.14}$$

$$x_{i,r}^{ps}[t] \in R^{+} \tag{3.15}$$

A parallel machine shop is a single process shop where the jobs can be processed in any of the machines in the shop floor. Different manufacturing set-up may have parallel machine shop, e.g., parallel machine flow shop and parallel machine job shop. However, in this research a single process parallel machine shop is considered where the different processes are independent. This manufacturing set up is used in continuous and batch processing such as steel plants and pharmaceuticals. In single process parallel machine shop, the set of orders $o \in \{1, 2, ..., \} = O$ to be processed in a day is known before the start of the shift. These jobs can be processed in any of the machine j in machine set J. All the jobs must be finished during the shift of the day where t_s is the shift start time and t_e is the shift ending time. s_{jk} is the time to process job k in machine j and it is assumed to be integer multiples of the time slot as we model the problem as discrete time model.

Production line loads satisfy the constraints in Equation 3.16 $-$ Equation 3.20. $a_{j,o}[t] = 1$ if job o is processed in machine i. $\alpha_{j,o}$ is the power required to process job k in machine j. β_o is the spike in power when machine j is switched on. Equation 3.16 ensures that all the jobs are completed within the day shift. Equation 3.17 ensures non interruptible nature of the job such that a job is not removed from the machine until it is processed. Equation 3.18 ensures that only one job is being processed in a machine at a given time. Equation 3.19 states the power requirement ($PR_{j,o}[t]$) and Equation 3.20 is the variable type constraint. First part of $PR_{j,o}[t]$ is the power required to process a job and second part is the spike in power due to the switching on of a machine when job o starts in machine j at time t.

$$\sum_{t=t_s}^{t_e}\sum_{o\in J} l_{j,o}[t] = s_{j,o} \qquad \forall o \in O \tag{3.16}$$

$$l_{j,o}[t] \leq l_{j,o}t^{'} \qquad \forall o \in O, \forall j \in J, \forall t^{'} \leq t + s_{j,o} \tag{3.17}$$

$$\sum_{o\in O} l_{j,o}[t] = 1 \qquad \forall j \in J, \forall t \in T \tag{3.18}$$

$$PR_{j,o}[t] = \alpha_{j,o} l_{j,o}[t] + \beta_j (l_{j,o}[t] - l_{j,o}[t-1]) \qquad \forall o \in O, \forall j \in J, \forall t \in T \tag{3.19}$$

$$l_{j,o}[t] \in \{0, 1\} \quad \forall j \in J, \forall k \in K, \forall t \in T \tag{3.20}$$

Data for the electrical loads for the different load types used in this chapter is given in Table 3.2.

Table 3.2.
Load requirements of different load types

Load	Residential house	Commercial building	Factory
Time shiftable load	U(2,4) kW	U(8,15)kW	U(8,15)kW
Modifiable power load	U(2,4) kW	U(8,15)kW	U(8,15)kW
Production line load	-	-	U(30,50)kW

Hourly electricity prices are shown in Figure 3.1. $p_2[t]$ is taken from [258] and $p_1[t]$ is obtained from $p_2[t]$ such that $p_2[t] = c\frac{H[t]}{\sum_{t\in T} H[t]}$ where c is a constant.

A case study on deterministic scheduling is presented in Section 3.1 and dynamic load scheduling using an online algorithm is presented in Section 3.2.

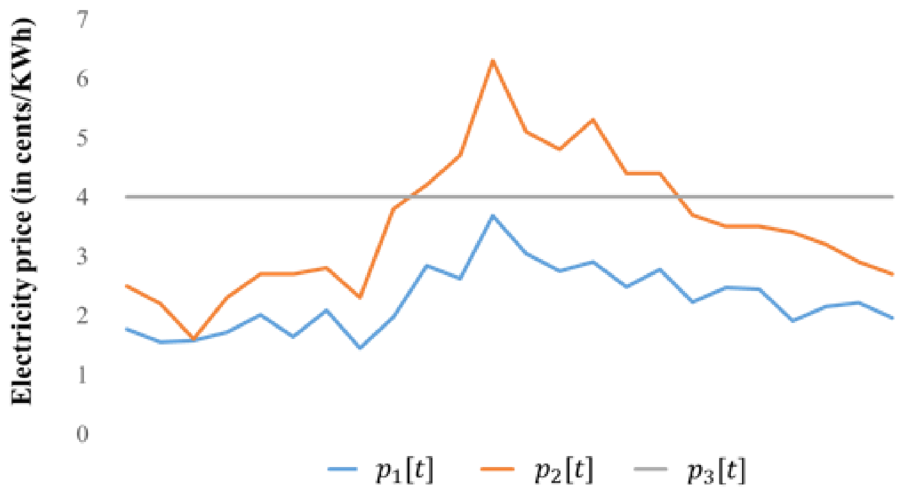


Fig. 3.1. Electricity hourly prices

3.1 Static scheduling using mixed integer linear programming

In this section, a deterministic MILP model is constructed for load scheduling in a μG. Different loads can be served by the electricity harvested from renewables, base load plant in the μG, natural gas based peaker plants and electricity bought from the Macrogrid. The objectives of this model include:

- develop a formulation for integrating industrial load scheduling with other types of loads and consumers in a μG.
- develop a framework for analyzing storage requirement under different pricing policies and storage efficiencies based on the total electricity cost
- integrate supply side with demand side for achieving overall energy efficiency in a μG by scheduling electricity generation from peaker plants

Most of the works considered charging the battery from excess harvest ([259] and [190]). In this section, 2 different battery charging options have been considered. First option is charging the battery from excess harvest and second option is charging the battery from excess harvest as well as from the electricity bought from Macrogrid. The impact of storage sizing (storage capacity) is also investigated.

To understand the impact of storage sizing on total electricity cost, two different pricing models have been considered. First is TOU tariffs and second is CPP where the electricity prices are assumed to be known. PAR is also studied as it is critical for the reliability of any grid.

3.1.1 Mathematical model

The load scheduling model is constructed to minimize the total electricity cost in a μG in a day. The model is called long term scheduling as the decisions are made in a single time period for a long time period. The μG decisions integrate load supply side decisions and demand side decisions, thus this is an integrative model. Let $Ha_w[t]$ and $Ha_s[t]$ be harvested from a wind and a solar farm at time t. Electricity from the renewables used at time t is $X[t] \leq Ha[t]$ where $Ha[t] = \sum_{w \in W} Ha_w[t] + \sum_{s \in S} Ha_S[t] \quad \forall t \in T$. The μG may run peaker plant at level e where $pp_{f,e}[t] = 1$ if peaker plant f runs at level e generating $P_{f,e}$ kW at time t. 8 different models are constructed to understand the investigate the trade-offs between electricity cost with different storage capacity under different pricing schemes and storage efficiency. The models are shown in Table 3.3. in Model V and Model VI, loads are scheduled randomly while loads are scheduled at the time of minimum cost in Model VII and Model VIII.

The objective function of minimizing the total electricity cost under TOU is given in Equation 3.21 and under CPP is given in Equation 3.22. In this section, we consider that time interval between two slots is 15 minutes. Therefore, Equation 3.21 and Equation 3.22 include a factor of 4. Total load at time t is given in Equation 3.23 and the peak load is stated in Equation 3.24. Supply and demand matching is shown in Equation 3.25. Storage constraints, time shiftable load constraints, modifiable power load constraints and production line load constraints have been explained earlier and

Table 3.3.
Different load scheduling models in Chapter 3

Model	Pricing	Charging: Excess harvest	Charging: Macrogrid
Model I	TOU	Yes	Yes
Model II	TOU	Yes	No
Model III	CPP	Yes	Yes
Model IV	CPP	Yes	No
Model V	TOU	Yes	No
Model VI	CPP	Yes	No
Model VII	TOU	Yes	No
Model VIII	CPP	Yes	No

stated in Equation 3.26, Equation 3.27, Equation 3.28 and Equation 3.29 respectively. Variable type constraint for different decision variables are shown in Equation 3.30 and Equation 3.31.

$$minimize \quad \Big(\sum_{t\in T}(p_1[t]X[t] + p_2[t]G[t] + \bar{c}[t]cd + \sum_{f=1}^{F}\sum_{e=1}^{E} P_{f,e})\Big)/4 \tag{3.21}$$

$$minimize \quad \big(\sum_{t\in T}(p_1[t]X[t] + p_2[t]G[t] + \bar{c}[t]cd + \sum_{f=1}^{F}\sum_{e=1}^{E} P_{f,e} + cpZ)\big)/4 \tag{3.22}$$

$$L[t] = \sum_{i\in I} L_i^{ns}[t] + \sum_{i\in I}\sum_{r\in R} L_{i,r}^{ts} x_{i,r}^{ts} + \sum_{i\in I}\sum_{h\in H} x_{i,h}^{ts} + \sum_{j\in J}\sum_{o\in O} PR_{j,o}[t] \quad \forall t \in T \tag{3.23}$$

$$Z \leq L[t] \quad \forall t \in T \tag{3.24}$$

$$G[t] + X[t] + B[t] = Y[t] + L[t] \quad \forall t \in T \tag{3.25}$$

$$Equation\ 3.1 - Equation\ 3.8 \tag{3.26}$$

$$Equation\ 3.9 - Equation\ 3.11 \tag{3.27}$$

$$Equation\ 3.12 - Equation\ 3.15 \tag{3.28}$$

$$Equation\ 3.16 - Equation\ 3.20 \tag{3.29}$$

$$G[t], X[t], A[t], B[t], Y[t], x^{ps}_{i,h}[t] \in R^{+} \quad \forall t \in T, \forall i \in I, \forall h \in H \tag{3.30}$$

$$x^{ts}_{i,r}[t], pp_{f,e}[t], \bar{c}[t] \in \{0, 1\} \quad \forall t \in T, \forall i \in I, \forall r \in R, \forall f \in F, \forall e \in E \tag{3.31}$$

In Model II and Model IV, battery is charged from excess harvest in the μG only and not from Macrogrid. Equation 3.32 enforces the constraint on battery charging from excess harvest only.

$$Y[t] \leq (X[t] - L[t])^{+} \quad \forall t \in T \tag{3.32}$$

Peak to average ration (PAR) is calculated as shown in Equation 3.33. PAR depends on the pricing policy adopted in the market and the storage capacity of the μG as shown in Section 3.1.2. Lower PAR is preferred as it requires lesser infrastructure (e.g. transmission and distribution lines) cost to build a system of lower capacity.

$$PAR = \frac{max_{t \in T}\big(G[t] + X[t] + \sum_{f \in F} \sum_{e \in E} P_{f,e}\big)}{average_{t}\big(G[t] + X[t] + \sum_{f \in F} \sum_{e \in E} P_{f,e}\big)} \tag{3.33}$$

3.1.2 Results

The μG has 150 houses, 4 commercial building and 1 factory. To study storage capacity requirement, 3 different battery efficiencies have been considered - 1, 0.8 and, 0.6. 3 peak pricing costs *cp* are considered - \$0.4, \$1 and, \$1.6. The trade-off between the total electricity cost and storage option under different battery efficiency and TOU tariffs is shown in Figure 3.2. It is observed as expected that as the storage capacity increases, the total cost decreases. Also, as the storage efficiency decreases, total electricity cost increases. The trade-off between the total electricity cost and storage option under different battery efficiency and CPP is shown in Figure 3.3.

Figure 3.4(a) shows the PAR under TOU pricing for the different models. It is observed that the PAR is highest for individualistic scheduling (Model VII) and lowest for random scheduling (Model V). PAR is minimum for Model V because we generate the time windows for loads using uniform distribution and therefore loads are distributed uniformly, thus, minimizing PAR. For Model I and Model II, PAR is in between the individualistic and random scheduling. Also, PAR increases as storage

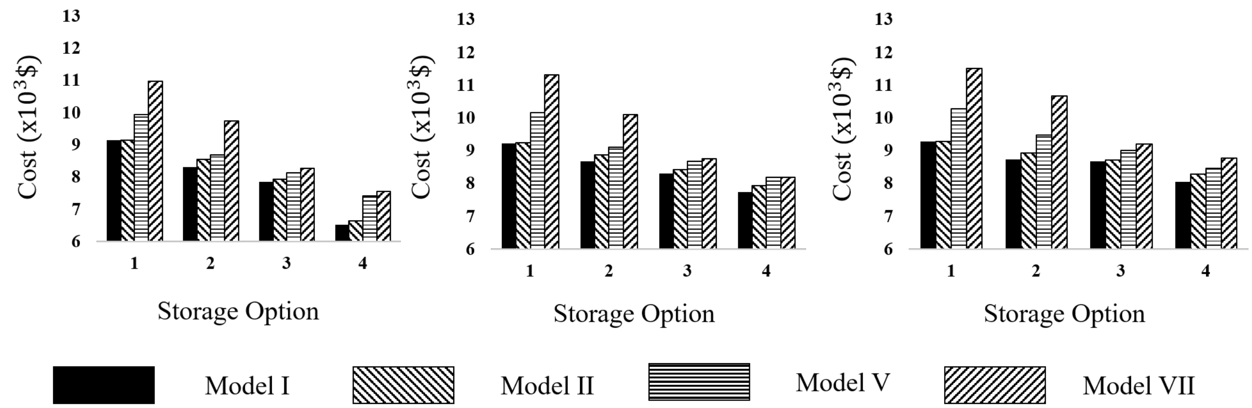


Fig. 3.2. Electricity cost for different storage options under TOU pricing

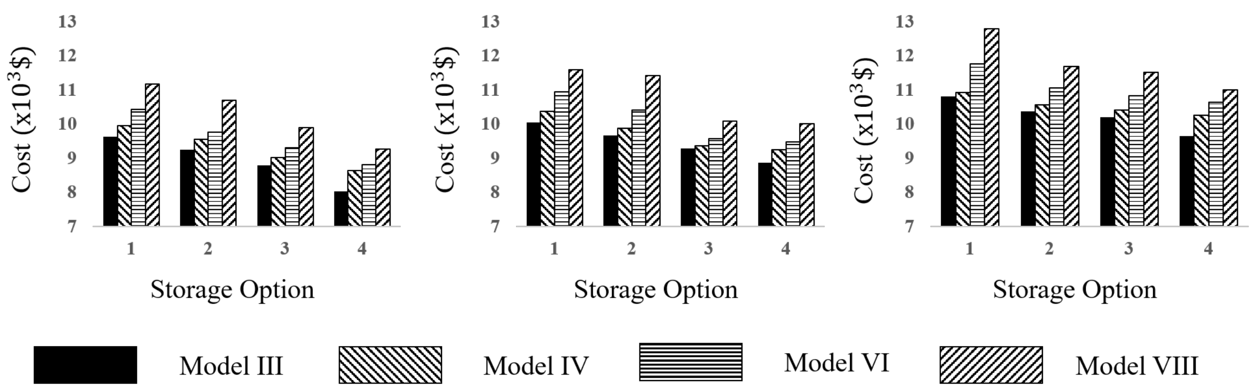


Fig. 3.3. Electricity cost for different storage options under CPP

capacity increases because more loads are scheduled at the time of minimum cost which can be met through storage. PAR for Model V and Model VII show no pattern with storage capacity as it is independent of the storage capacity. Figure 3.4(b) and Figure 3.4(c) show PAR for Model III and Model IV respectively under the different peak prices. In this experiment, storage efficiency is considered to be 0.8. It shows that as the peak price increases, PAR decreases and loads may be scheduled at the time of lower electricity costs.

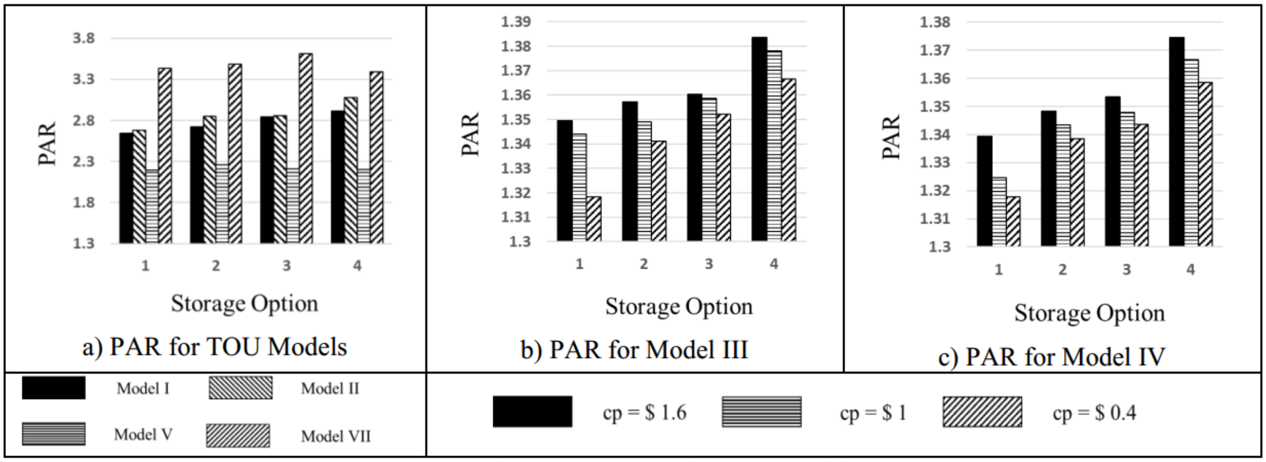


Fig. 3.4. PAR for different models

3.1.3 Conclusion and discussion

In this model, deterministic load scheduling in a μG is addressed. The μG is considered as a network of different (types of) generators serving different (types of) demands by different (types of) consumers. Different models are analyzed based on two pricing policies under two conditions of how the battery is charged. A mixed integer programming model is constructed. The results show the trade-offs between the total electricity cost with different storage capacity and storage efficiency. It is observed that charging battery from the Macrogrid allows more flexibility in scheduling loads thus total electricity cost is lowered. Results also show that the Peak to Average Ratio (PAR) is highest for individualistic scheduling and lowest for random scheduling. Electricity cost decreases as the storage capacity increases. PAR increases as the storage capacity increases as more jobs can be scheduled at the same time. As the peak price increases, PAR decreases for optimal scheduling. The analysis provided on the storage capacity and PAR could help in designing storage sizing based on the pricing policy in a μG. This is one of the first works in storage sizing which accounts for storage efficiency.

Even though the problem discusses a simplified deterministic case, the paper presents rigorous analysis on storage sizing, PAR and, pricing policies that will serve as a guideline and benchmark for practical situations with uncertain electricity harvest, demand and, real time decision making. The model could be studied for different manufacturing set-up with different production line loads. In this section, it is assumed that DLCs send exact information to the SO of different loads to be scheduled.

In the deterministic model, since the decisions are made assuming that the information is deterministic, the optimal decisions are made for a longer time period. In the next model, real-time decisions are made for a short term based on the current information. Since the decisions are made based on current information, future events are not considered in these models. This might lead to sub-optimal decisions.

3.2 Dynamic load scheduling using Lyapunov optimization

The load scheduling is modeled as a dynamic scheduling model with N consumers and a central controller as shown in Figure 3.5. DLCs represent the consumers and SO represents the controller. The model is called dynamic scheduling model because it schedules loads over a sequence of time slots $t \in T = \{0, 1, 2, ..., |T|\}$ and there are random events in every time slot. The random events include electricity harvest from renewables, electricity price and, dynamic requests from consumers. The controller observes the random event vector but the consumers observe only their respective random event. The objectives of this Section include:

- develop a dynamic model for integrating industrial load scheduling with other types of loads and consumers in a μG.
- develop a framework for analyzing storage requirement under real - time pricing policy

Most of the works as mentioned in Section 2 consider static scheduling. In real time electricity markets, electricity prices are released 15 minutes prior to the time. Also, a better estimate of electricity harvest from renewables can be made closer to the time. The idea to use real time information for optimal scheduling provides motivation for building a dynamic load scheduling model.

Since the loads are scheduled at different time slots with different electricity prices, some loads are scheduled at the time of higher prices. To ensure consumer participation in coordinating with the controller, fairness is used such that a consumer pays in accordance with her total consumption as shown in Equation 3.36. For example, let TC_{i1} and TC_{i2} are the total electricity cost paid by residential consumer $i1$ and $i2$ respectively and rest of the terms in Equation 3.36 are explained in the previous Section. F_i is the flexibility given by a consumer in terms of time flexibility in load scheduling to the controller. Flexibility for the industrial consumer is given by Equation 3.34 while for non industrial consumer is given by Equation 3.35.

$$F_i = \sum_{r \in R} (t_{i,r}^{ts,max} - t_{i,r}^{ts,max}) + \sum_{h \in H} (t_{i,h}^{ps,max} - t_{i,h}^{ps,max})$$
$$+ \sum_{k \in K} (t_e - t_s) \ \forall i \mid i \text{ is industrial consumer} \tag{3.34}$$

$$F_i = \sum_{r \in R} (t_{i,r}^{ts,max} - t_{i,r}^{ts,max}) + \sum_{h \in H} (t_{i,h}^{ps,max} - t_{i,h}^{ps,max})$$
$$\forall i \mid i \text{ is non industrial consumer} \tag{3.35}$$

$$\frac{TC_{i1} F_{i1}}{TC_{i2} F_{i2}} = \frac{\sum_{t \in T} T_{i1}^{ns}[t] + \sum_{r \in R} L_{i1,r}^{ts} + \sum_{h \in H} L_{i1,h}^{ps}}{\sum_{t \in T} T_{i2}^{ns}[t] + \sum_{r \in R} L_{i2,r}^{ts} + \sum_{h \in H} L_{i2,h}^{ps}} \tag{3.36}$$

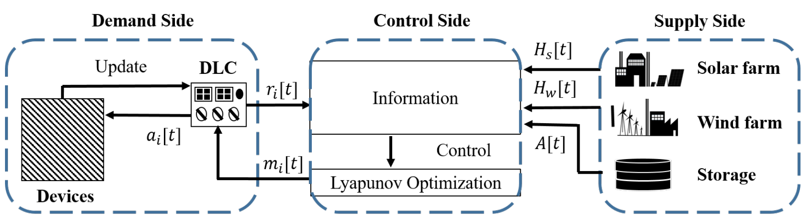


Fig. 3.5. Schematic representation of a dynamic scheduling model

In every time slot t, consumer i sends request $r_i[t]$ to the controller. Thus, the controller observes the request vector $\{r_1[t], r_2[t], ..., r_N[t]\}$. It takes control action based on the online algorithm explained in Section 3.2.2 to find optimal message vector $\{m_1[t], m_2[t], ..., m_N[t]\}$. Based on the message vector, the consumer schedules loads through action $a_i[t]$. As assumed that the consumers coordinate in dynamic scheduling, $a_i[t] \leq m_i[t], \forall i \in I$, $\forall t \in T$. The action message $a_i[t]$ determines which load is served by the DLC of player i.

First, a benchmark model is developed in Equation 3.37 $-$ Equation 3.26 that assumes complete information of the various stochastic variables at the start of the day such as real-time electricity price, electricity harvest from the renewables, and consumer demand. This benchmark model is constructed with the aim of obtaining an optimal solution for the dynamic scheduling model discussed later in this Section.

3.2.1 Benchmark Model

Total electricity cost is shown in Equation 3.37. The storage constraints and load constraints for different load types are shown in Equation 3.38 − Equation 3.41. Total load at time t is given by Equation 3.42 while supply and demand balance is shown in Equation 3.43. Equation 3.44 and Equation 3.44 are the variable type constraint.

$$\text{Min} \qquad \sum_{t \in T} \big(X[t]p_1[t] + G[t]p_2[t] + \bar{c}[t]cd \big)/4 \tag{3.37}$$

$$Equation\ 3.1 - Equation\ 3.8 \tag{3.38}$$

$$Equation\ 3.9 - Equation\ 3.11 \tag{3.39}$$

$$Equation\ 3.12 - Equation\ 3.15 \tag{3.40}$$

$$Equation\ 3.16 - Equation\ 3.20 \tag{3.41}$$

$$L[t] = \sum_{i \in I} L_i^{ns}[t] + \sum_{i \in I} \sum_{r \in R} L_{i,r}^{ts} x_{i,r}^{ts} + \sum_{i \in I} \sum_{h \in H} x_{i,h}^{ts} + \sum_{j \in J} \sum_{o \in O} PR_{j,o}[t] \quad \forall t \in T \tag{3.42}$$

$$G[t] + X[t] + B[t] = Y[t] + L[t] \quad \forall t \in T \tag{3.43}$$

$$G[t], X[t], B[t], Y[t], x_{i,r}^{ps}[t] \in R^+ \in \{0, 1\} \quad \forall t \in T \tag{3.44}$$

$$x_{i,r}^{ts}[t], a_{j,o}[t] \in \{0, 1\} \quad \forall t \in T \tag{3.45}$$

However, this model can be used only as benchmark for other models as electricity harvest from renewables and consumer demands can be realized only in real-time. This model could be solved at the end of the time horizon to check how well the proposed could have done. Dynamic models schedule the loads based on most recent and accurate information. Next , a step wise methodology to construct the dynamic scheduling model that uses an online algorithm is discussed.

3.2.2 Control Actions using Lyapunov optimization

In this paper, a real time model is considered with no assumption on future and all the decisions are made based on the current information. The dynamic scheduling model aims at minimizing the total electricity cost for the consumers by solving a greedy online scheduling problem at every time slot t. These greedy solutions will schedule no loads (except non-shiftable load which have to be served) to minimize the total electricity cost at time slot t. Thus a control parameter $Th_i \, , \forall i \in I$ is developed which forces the online problem to schedule some loads based on different weights given to the loads at time t. First, the construction of control parameter is discussed in Section 3.2.2 followed by the Lyapunov optimization based online algorithm in Section 3.2.3.

Control parameter

A threshold Th_i is used to force the greedy online scheduling problem to schedule some loads. If the flexible loads are not scheduled till their latest finish time, it has to be served in the next time slot, thus making it behave like a non-shiftable load that loses its flexibility. Thus, priority is given to the loads which were requested earlier. This a simple priority rule and different models can be build on different priority rules. Since the non-shiftable loads have to be served, it does not play a part in the threshold. The weights given to the different load types are defined in Table 3.4. $z_j[t] = 1$ if machine j is running at time t hence $z_j[t] = 1$ if $\sum_{k \in K} a_{j,k}[t-1] = 1$. The threshold function at time t for consumer i is shown in Equation 3.46 when consumer i is a factory. For residential and commercial consumers, threshold function is given in Equation 3.47.

$$\begin{aligned} T_i[t] = \sum_{r \in R} L^{ts}_{i,r}(1 - x^{ts}_{i,r}) w^{ts}_{i,r}[t] + \sum_{h \in H} (L^{ps}_{i,h} - x^{ps}_{i,h}) w^{ps}_{i,h}[t] \\ + \sum_{j \in J} \sum_{o \in O} (1 - l_{j,o}[t]) \big(\alpha_{j,o} + \beta_j z_j[t-1] \big) w^{pl}_{j,o}[t] - Th_i \end{aligned} \tag{3.46}$$

Table 3.4.
Variable definitions

Variable	Definition
$w^{ts}_{i,r}[t]$	$t - t^{ts,min}_{i,r} \mid t \geq t^{ts,min}_{i,r}, \sum_{t=t^{ts,min}_{i,r}}^{t-1} x^{ts}_{i,r}[e] = 0$
$w^{ps}_{i,h}[t]$	$t - t^{ps,min}_{i,h} \mid t \geq t^{ps,min}_{i,h}, \sum_{e=t^{ps,min}_{i,h}}^{t-1} x^{ps}_{i,r}[e] < L^{ps}_{i,r}$
$w^{pl}_{j,k}[t]$	$t - t_s, \mid t \geq t_s, \sum_{t=t_s}^{t-1} \sum_{o\in O} a_{j,o}[t] = 0$

$$T_i[t] = \sum_{r\in R} L^{ts}_{i,r}(1 - x^{ts}_{i,r})w^{ts}_{i,r}[t] + \sum_{h\in H}(L^{ps}_{i,h} - x^{ps}_{i,h})w^{ps}_{i,h}[t] - Th_i \tag{3.47}$$

A dynamic set $SL_i[t]$ is constructed for the control actions. Since the control actions are taken in every time step, $SL_i[t]$ includes loads that have been requested but not yet served till time t. The set $SL_i[t]$ is defined as a 2-tuple $\{L^{sl}_{i,d}, w^{sl}_{i,d}[t]\}$ where $L^{sl}_{i,d}$ is the load to be considered and $w^{sl}_{i,d}[t]$ is the weight for threshold. $d \in SL_i[t]$ is given in Equation 3.48. Using $SL_i[t]$, an approximation for Equation 3.46 is constructed in Equation 3.49.

$$\begin{aligned} &\{L^{ts}_{i,r}, w^{ts}_{i,r}\} \subset SL_i[t] \quad \mid \quad d \text{ is time shiftable load} \\ &\{\frac{l^{ts}_{i,h}}{t - t^{ps,min}_{i,h}}, w^{ts}_{i,h}\} \subset SL_i[t] \quad \mid \quad d \text{ is modifiable power load} \\ &\{\left(\frac{\sum_{j\in J}\alpha_{j,o}}{|J|}\right), w^{pl}_{j,o}\} \subset SL_i[t] \quad \mid \quad d \text{ is production line load} \end{aligned} \tag{3.48}$$

$$T_i[t] = \sum_{d\in SL_i[t]} L^{sl}_{i,d}w^{sl}_{i,d}(1 - x^{sl}_{i,d}) - Th_i \quad \forall i \in I, \forall t \in T \tag{3.49}$$

The intuition behind the threshold function is that the flexible loads should be served so as to minimize the delay in scheduling the loads. The model aims to minimize the time average value threshold function that is contradictory to cost function as electricity cost is incurred in scheduling the loads.

3.2.3 Lyapunov optimization

Lyapunov optimization relaxes the problem to time average minimization problem satisfying time average constraints. In every time slot t, the controller solves a greedy scheduling problem as discussed in this section. Objective function for minimizing the time average cost is shown in Equation 3.50 while time average constraint for the threshold function is given in Equation 3.51 where Th_i is a known constant.

$$\lim_{|T|\to\infty} \frac{1}{|T|} \sum_{t\in T} E\Big[X[t]p_1[t] + G[t]p_2[t] + \bar{c}[t]cd\Big] \tag{3.50}$$

$$\lim_{|T|\to\infty} \sum_{t\in T} \frac{1}{|T|} T_i[t] = \lim_{|T|\to\infty} \frac{1}{|T|} \sum_{t\in T} \sum_{d\in SL_i[t]} L^{sl}_{i,d} w^{sl}_{i,d}(1 - x^{sl}_{i,d}) - Th_i, \forall i \in I \tag{3.51}$$

A virtual queue Q_i for threshold function of consumer i from Equation 3.49 is developed as shown in Equation 3.52. If the virtual queue grows, the online algorithm schedules the load which aligns with the notion of minimizing the time to serve for the flexible loads.

$$Q_i[t+1] = \max\big(0, Q_i[t] + \sum_{d\in D} L^{sl}_{i,d} w^{sl}_{i,d}(1 - x^{sl}_{i,d}) - Th_i\big) \tag{3.52}$$

Lyapunov function is defined as $\omega[t] = \frac{1}{2}\sum_{i\in I}\big(Q_i[t]\big)^2$. A drift function is defined as $\triangle[t] = E\big[\omega[t+1] - \omega[t] | Q[t]\big]$ where $Q[t] = \{Q_1[t], Q_2[t], ..., Q_N[t]\}$. Equation 3.53 is derived from Equation 3.52 by taking the squares of both sides.

$$\big(Q_i[t+1]\big)^2 \leq \Big(Q_i[t] + \sum_{d\in D} L^{sl}_{i,d} w^{sl}_{i,d}(1 - x^{sl}_{i,d}) - Th_i\Big)^2 \tag{3.53}$$

Simplifying Equation 3.53 and using the drift function $\triangle[t]$, Equation 3.54 is derived. First part uses the facts that $(1-x) \leq x$ when $x \geq 0$ and $(a-b)^2 \leq (a+b)^2$ when $a, b \geq 0$.

$$\triangle[t] = \sum_{i\in N} \Big(\big(\sum_{d\in D} L^{sl}_{i,d} w^{sl}_{i,d} + Th_i\big)^2 + Q_i[t]\big(\sum_{d\in D} L^{sl}_{i,d} w^{sl}_{i,d}(1 - x^{sl}_{i,d}) - Th_i\big)\Big) \quad \forall t \in T \tag{3.54}$$

Let the constant $C = \sum_{i\in N}\Big(\big(\sum_{d\in D} L^{sl}_{i,d} w^{sl}_{i,d} + Th_i\big)^2 + Q_i[t]\big(\sum_{d\in D} L^{sl}_{i,d} - Th_i\big)\Big)$.

Theorem 1 The virtual queue Q_i is mean rate stable.

Proof The proof is provided in the Appendix **??**.

A drift+penalty function is formed in Equation 3.55 where a penalty term is added to account for the objective function of minimizing total electricity cost.

$$\nabla[t] \leq C|T| - \sum_{i \in I} Q_i[t] \sum_{d \in D} L^{sl}_{i,d} w^{sl}_{i,d} x^{sl}_{i,d} + V\big(X[t]p_1[t] + G[t]p_2[t]\big) \quad \forall t \in T \tag{3.55}$$

Intuitively, Lyapunov optimization aims at minimizing the average electricity cost while maintaining the stability of the virtual queues Q_i. V is the weight factor for the objective function which involves minimizing a bound on the drift and penalty as two contradicting functions. Increasing V gives greater priority to minimize the cost at the expense of greater size of the virtual queue.

SO takes control actions by solving the following Mixed Integer Linear Programming optimization problem in Equation 3.56 $-$ Equation 3.60 over each time slot t. These control actions aim to minimize the drift + penalty function by control actions in every time slot. The objective function is given in Equation 3.56. Supply and demand balance is shown in Equation 3.57. Storage constraints and feasibility constraint for a machine are introduced in the model in Equation 3.58 and Equation 3.59 respectively. Equation 3.60 and Equation 3.61 are the variable type constraint.

$$\text{Minimize} \ -\sum_{i \in I} Q_i[t] \sum_{d \in D} L^{sl}_{i,d} w^{sl}_{i,d} x^{sl}_{i,d} + V\big(G[t]p_2[t] + X[t]p_1[t]\big) \tag{3.56}$$

$$G[t] + X[t] + B[t] = Y[t] + \sum_{i \in I} L^{ns}_i[t] + \sum_{i \in I} \sum_{d \in D} L^{sl}_{i,d} x^{sl}_{i,d} \tag{3.57}$$

$$\text{Equation 3.1} - \text{Equation 3.8} \tag{3.58}$$

$$\sum_{o \in O} a_{j,o}[t] = 1 \quad \forall j \in J, \forall o \in O \tag{3.59}$$

$$x^{sl}_{i,d}, a_{j,o}[t] \in \{0,1\} \quad \forall i \in I, \forall d \in D, \forall j \in J, \forall o \in O \tag{3.60}$$

$$G[t], X[t], B[t] \in R^+ \quad \forall i \in I \tag{3.61}$$

After finding the decision variables in every time slot t, the virtual queues $Q_i[t]$ are updated based on Equation 3.53, shiftable load set $SL_{i,t}$ is updated based on the decision variable values $x^{sl}_{i,d}$ and, storage in battery $A[t]$ is updated based on Equation 3.1. If $x^{sl}_{i,d} = 1$ and the load takes more than one time slot, then it is considered as non-shiftable load for the next time slots running at the same power as at time t. This is a drawback of the proposed online algorithm which may lead to sub-optimal solutions. It can be alleviated by stochastic dynamic scheduling using estimation for future time slots which is considered in Chapter 4, Chapter 5 and Chapter 6.

The proposed model developed in this Section is a relaxed version of the benchmark model discussed in Section 3.2.1. The proposed model aims to minimize the total electricity cost as in the benchmark model. The storage constraints and production line constraints are ensured in the online scheduling problem solved in every time slot t. The load constraints are satisfied while constructing the dynamic set $SL_i[t]$. A flexible load becomes non-shiftable load if it not served till its latest finish time, thus the time window constraints are also followed in the proposed model.

3.2.4 Results

A μG with residential houses, commercial building and a factory with parallel machine shop is considered. μG meets the consumers' demand by harvesting electricity from renewables (solar farm and wind farm), discharging electricity from the battery and buying electricity from Macrogrid when required. The objective of the model is to minimize the total electricity cost in the μG for a day.

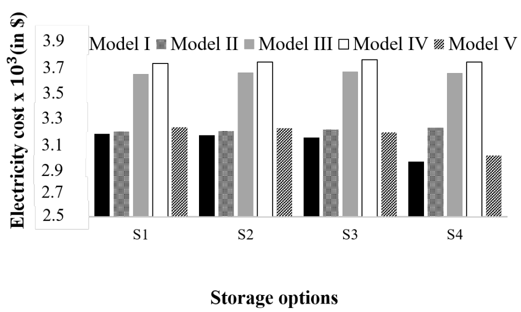
To verify the performance of the proposed dynamic scheduling model, its performance is compared with different models as discussed in Table 3.5. Model I is the benchmark model with perfect information of future events and provides benchmark solution. In Model II, loads are scheduled based on estimation of demand and electricity harvest using formulation in Section 3.2.1. This model is based on the strong assumption that all the loads are known at the start of the day that is impractical. This model is used for scheduling electricity generation in generating plants [42]. In Model III, loads are scheduled at the time of minimum cost in their time windows. This model is used in building control to minimize the total cost of a consumer as some of the works mentioned in Section 2. This model aims at optimization at a consumer level leading to overall increase in cost leading to peak demand at the time of minimum cost. It is assumed that prices are estimated for Model II and Model III. In Model IV, static strategy of scheduling the loads when requested is adopted. It is the simplest model that does not have any assumption and can be adopted in any electricity market.

Table 3.5.
Different load scheduling models

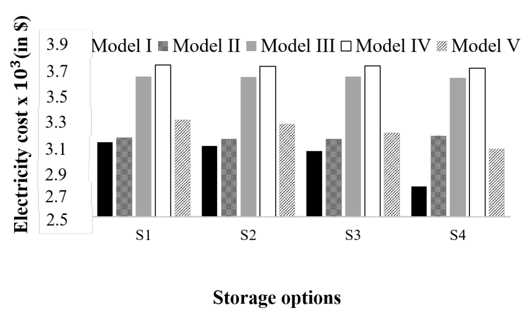
Model	Definition
Model I	Benchmark scheduling (Section 3.2.1)
Model II	Scheduling based on estimation
Model III	Individualistic scheduling
Model IV	Static strategy
Model V	Dynamic scheduling model

The results for the different methods is shown in Figure 3.6. Different ratios of non-shiftable to flexible loads are considered. This ratio of non-shiftable to flexible load is defined as load ratio in this research. It can be shown that as the load ratio increases, the total electricity cost increases as flexible loads provide the flexibility of load shifting. Also, as the storage capacity increases along the different options, total cost decreases. These trade-offs between total electricity cost and storage capacity can be used as a guideline in μG design based on Techno-economic analysis of savings from the different storage capacity.

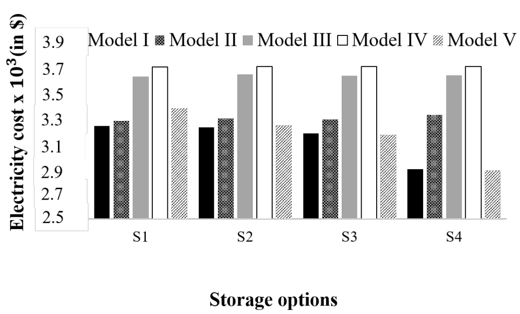
Individualistic scheduling performs the worst because all the loads are scheduled by the consumers independently of other consumers at the time of minimum cost, thus creating a spike in demand. This spike is met by expensive electricity from the Macrogrid. This confirms that local information optimization performs poorly against global information optimization. Static strategy of scheduling loads when requested is also expensive as compared to the dynamic model proposed in this paper. This is because the flexibility of the loads is not leveraged and loads are scheduled at the time of peak consumption. Scheduling based on estimation performs comparatively to the benchmark but its assumptions are impractical. The estimation based scheduling model can be improved and made dynamic by rescheduling after every information. Instead of reactive scheduling, the proposed dynamic scheduling model schedules the load based on current information. The model is practical, has minimum assumptions and performs comparatively well as against the optimal benchmark solution. The results are obtained for the parameter value $V = 10$ assuming $Th_i = 50$ for residential consumers, $Th_i = 100$ for commercial consumers and $Th_i = 250$ for the industrial consumers. The value of Th_i is obtained such that on an average, the flexible loads are served by half of its time window.



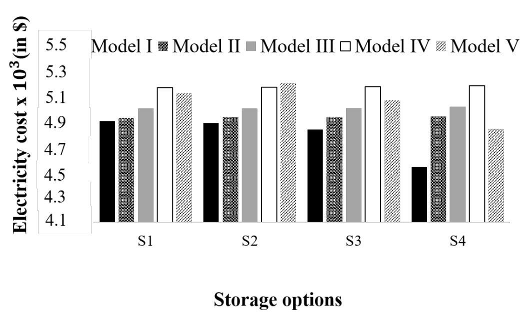
(a) Ratio = 0.5



(b) Ratio = 1



(c) Ratio = 2



(d) Ratio = 8

Fig. 3.6. Trade-offs between electricity cost and storage

The proposed dynamic scheduling model performs comparatively to the benchmark solution. Table 3.6 shows the optimality gap in the solution of the proposed model as compared to the benchmark solution. For most of the cases, the error is within acceptable 5%.

Table 3.6.
Optimality gap in the proposed model as compared to benchmark solution

Storage option	Error (in %)
S1	3.1
S2	2.7
S3	2.0
S4	3.5

A mathematical study on optimal value of V will be considered in future work. The performance of the algorithms has been done using $V = 10$. Different experiment run show that the algorithm is robust to the value of V and a mathematical study on optimal value of V will be considered in the future work. The algorithm performs bad when $V = 0$ and when $V = \infty$ as in the former case, the algorithm aims to decrease the drift hence scheduling all the job when requested. In the later case, the algorithm does not schedule any load and all the loads are scheduled after being converted as non-shiftable loads.

An example of optimal load profile for Model I and Model V is shown in Figure 3.7. The results have been obtained for load ratio = 2, storage option 3 and, $V = 10$. It can be seen that Model I maintains the number of loads scheduled at a given time since the demand data is generated uniformly. There is spike is load profile when the electricity price is lower. Model V on the other hand I shows a lag in the load profile as compared to the load profile for Model I. This is because the dynamic model waits to schedule the load based on the drift+penalty function and some of the loads are scheduled at the time of their latest finish time. Total harvest from generation is also shown in Figure 3.7 to show that electricity from renewables is generally out of sync with the demand and storage can be used to shift the usage across time.

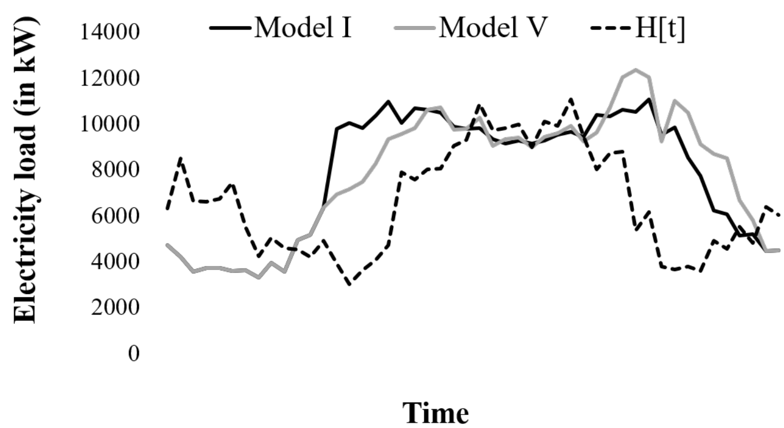


Fig. 3.7. An example of load profile

However, in this Section, as the load ratio decreases from 2 to 0.5, the cost increases for the test case shown in Figure 3.6(a). This is because of the methodology by which the test cases are generated. A multiplication factor is used for non-shiftable and shiftable loads and as the ratio is decreased, demand for non-peak hours decreases as shown in Figure 3.8 - Figure 3.10 while demand for flexible loads during peak hours increases. Thus as the ratio decreases beyond a certain point, the density of demand intensifies during a given time period, increasing the consumption of expensive electricity from Macrogrid.

Peak-Average Ratio (PAR) is a critical measure for system reliability. It is defined as Peak load in a given time horizon by average load. It can be seen from Figure **??** that as storage capacity increase, PAR increases as more loads can be scheduled at a given time. This confirms that increasing the storage capacity can hedge against spike in demand due to unforeseen events such as machine failure. However, building and maintenance of storage is expensive thus storage sizing in a μG is critical.

μGs could also sell $S[t]$ electricity at time t back to the Macrogrid when it is more profitable to sell the electricity. The objective functions in Section 3.2.1 and

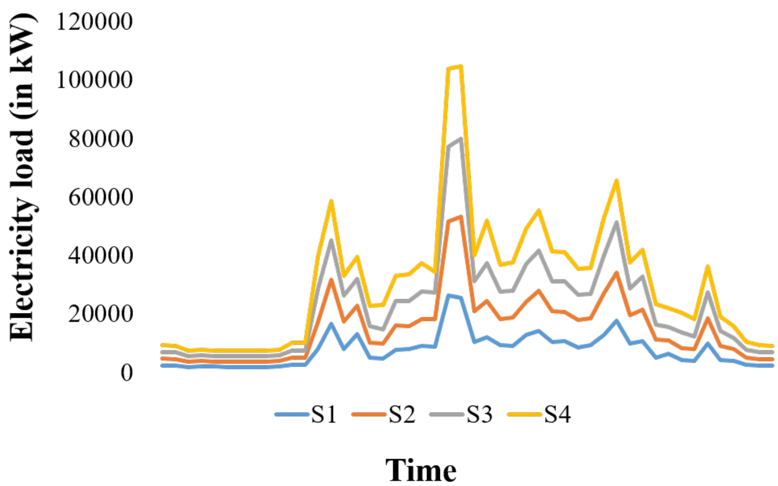


Fig. 3.8. Demand profile under Load ratio = 0.5

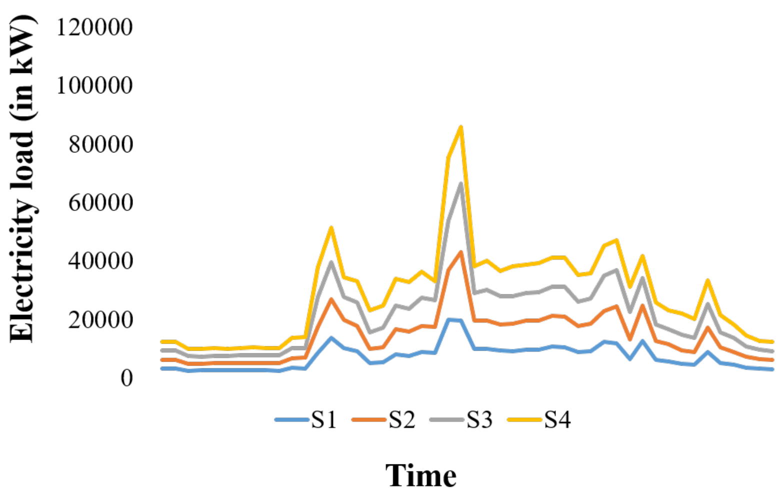


Fig. 3.9. Demand profile under Load ratio = 1

the penalty term in Section 3.2.2 are updated as shown in Equation 3.62 and Equation 3.63 to account for selling the electricity back to the Macrogrid. Also, Equation 3.24 and Equation 3.56 are updated as shown in Equation 3.64 and Equation 3.65. The results for Model I and Model V are shown in Figure 6.4 where $p_3[t] = 1.5p_2[t]$.

$$\text{Min} \qquad \sum_{t \in T} X[t]p_1[t] + G[t]p_2[t] - S[t]p_3[t] \tag{3.62}$$

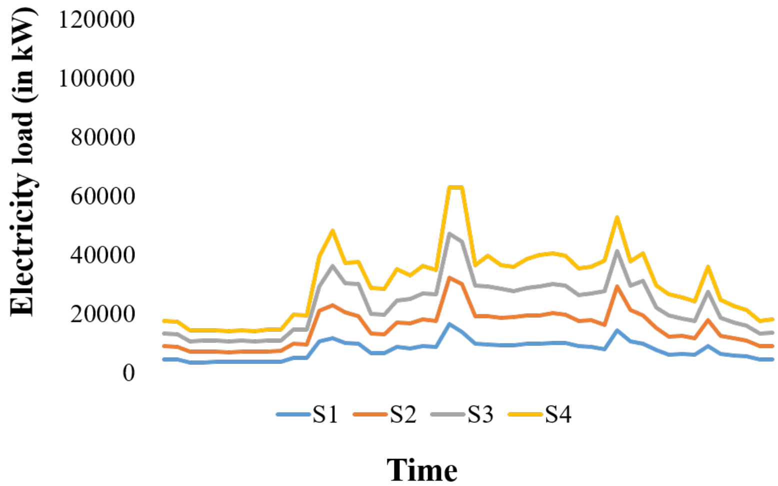


Fig. 3.10. Demand profile under Load ratio = 2

$$\text{Min} \quad -\sum_{i\in I} Q_i[t]\sum_{d\in D} L^{sl}_{i,d}x^{sl}_{i,d} + V\big(G[t]p_2[t] + X[t]p_1[t] - S[t]p_3[t]\big) \tag{3.63}$$

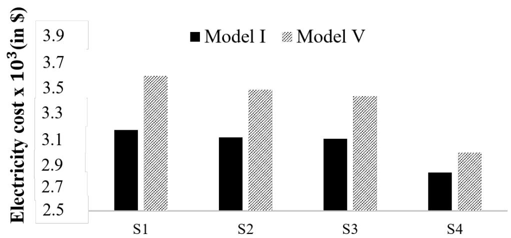
$$G[t] + X[t] + B[t] = Y[t] + L[t] + S[t] \quad \forall t \in T \tag{3.64}$$

$$G[t] + X[t] + B[t] = Y[t] + S[t]\sum_{i\in I} L^{ns}_i[t] + \sum_{i\in I}\sum_{d\in D} L^{sl}_{i,d}x^{sl}_{i,d} \tag{3.65}$$

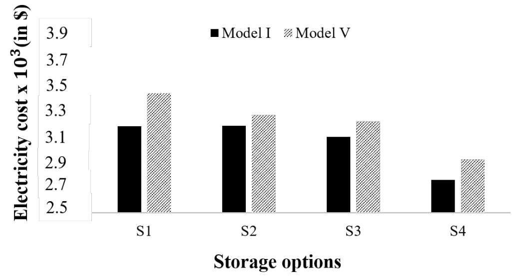
The performance of the proposed model as against benchmark solution in shown in Table 3.7 when μG could sell electricity back to the Macrogrid. The proposed model performs within approximately 5% error of the benchmark solution.

Table 3.7.
Optimality gap when μG could sell electricity to the Macrogrid

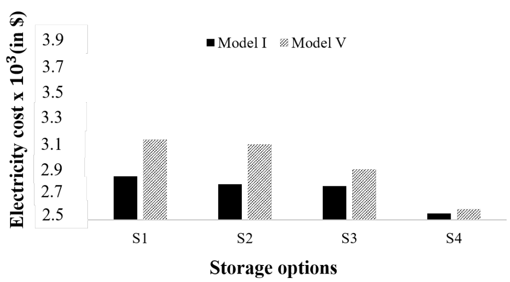
Storage option	Error (in %)
S1	5.4
S2	5.2
S3	4.2
S4	2.8



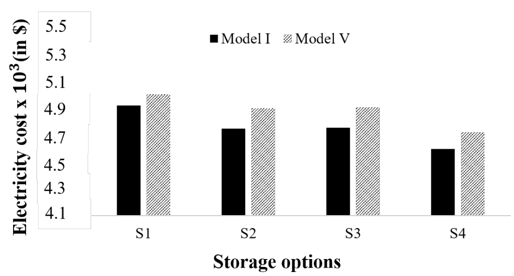
(a) Ratio = 0.5



(b) Ratio = 1



(c) Ratio = 2



(d) Ratio = 8

Fig. 3.11. Electricity cost for different storage options when μG could sell electricity

The load profile for Model I and Model V is shown in Figure 3.12 for storage option 3 and load factor 2 when μG can sell electricity to the Macrogrid. As shown in Figure 3.7, loads profile is approximately uniform for Model I. Load profile in Model V tries to match the profile for non-shiftable loads. There are spikes when the drift function outweigh the penalty term. The proposed model illustrates that the future scenarios of consumers selling back electricity could be included in the model.

Computation time is an important measure for the performance of the algorithm. Particularly, when production lie loads are considered, most of the problems are NP-hard in nature. Since the proposed algorithm solves sub-problems in every discrete time, the computation time is expected to be low as against Model I and Model II. The average time to solve the problem using the proposed framework is shown in Table 3.8. It confirms lower computation time for the proposed framework. The results show that as the number of flexible load decreases (load ratio increases), computation time

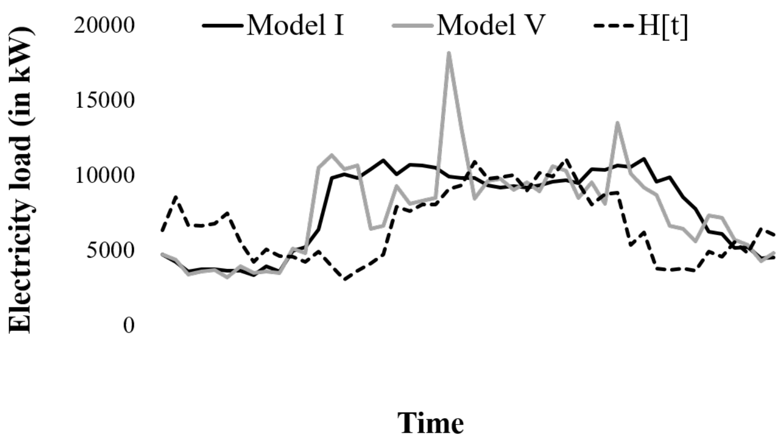


Fig. 3.12. Example of load profile when μG could sell electricity

Table 3.8.
Computation time (in seconds) for different load ratio

Load Ration	Model I	Model V
0.5	30.02	0.69
1	21.42	0.45
2	13.67	0.34
8	5.45	0.22

decreases as the number of decision variables and constraints decreases. Thus, the proposed model is feasible as the decision variables lie in feasible set of the original problem in Section 3.2.1.

Operating in islanded state is an important property of the μGs. Storage Option 1 and Storage Option 4 are studied in terms of the number of instances when G withdraws electricity from Macrogrid. The results are shown in Table 3.9. It is observed that on increasing the storage size, μG stays in islanded state more frequently. Difference in proportion in rows 1 and 2 indicate that electricity was drawn from the Macrogrid just to charge the storage. In dynamic scheduling, the model draws electricity from Macrogrid more frequently as decisions are made without any information of the future.

Table 3.9.
Proportion of time slots when μG work in Islanded state

Model	Storage option 1	Storage option 4
I,III	0.542	0.605
II,IV	0.563	0.667
Dynamic scheduling	0.332	0.383

To study the computational time of the model, model for different problem sizes have been solved. The problem size is shown with (a1, a2, a3) where a1 is the number of industrial, a2 is the number of commercial and, a3 is the number of residential consumers. The average computational time is shown in Table 3.10. Computation is higher for model where electricity is not drawn from the Macrogrid because of the added constraint in the model. It is observed that as the problem size increases, the computational time increases exponentially for deterministic model. Since the dynamic model runs the optimization model for every time slot, time taken to solve the problem is in the order of 10 seconds that proves that the model is scalable for bigger problem instances. Dynamic model obtains results with around 7% error as the decisions are made without future information and sub-problems are solved in every time slot t.

3.2.5 Conclusion and discussion

An automated load scheduling framework under the umbrella of dynamic scheduling was presented to deal with efficient energy management in a μG. The framework uses an online algorithm where the central controller take control actions based on Lyapunov optimization. It integrates different (types of) loads from different (types of) consumers. This Section also studies storage sizing and impact of ratio of non-

Table 3.10.
Computational time (in seconds) for different problem size

Problem size	Model I, III	Model II, IV	Dynamic scheduling
(1, 3, 150)	40.32	120.65	1.37
(1, 3, 150)	40.32	120.65	1.37
(2, 6, 300)	120.34	481.21	2.34
(3, 9, 450)	250.67	1085.19	4.69
(4, 12, 600)	480.53	2208.34	6.92
(5, 15, 750)	670.23	3986.08	13.67

shift able loads to flexible loads. It is one of the first work that considers integrating production line load with different consumer types for dynamic energy management in a μG. Comparative analysis of the proposed dynamic model with different realistic models confirms that the proposed model outperforms other models and it obtains results within 5% error of the benchmark solutions.

The model can be extended in different directions. This Chapter does not investigate a rule for selecting the optimal values of the weight parameter V which can be obtained by extensive simulation and experimental analyses. However, the results show that for the given test cases, the model is fairly robust to the values of V. However built in this Chapter does not ensure consumer privacy as all the information is sent to the controller, that is, the SO. Chapter 5 includes constructing an intelligent algorithm for DLCs for dynamic load scheduling such that all the scheduling decisions for a consumer are made by the algorithm itself and it shares partial information, or just requests to the SO.

4. INTELLIGENT ALGORITHM FOR DYNAMIC LOAD SCHEDULING IN A JOB SHOP

In the previous Chapter, a simple parallel machine shop was considered for the production line loads. Majority of continuous production factory e.g. steel processing and pharmaceuticals consider different processing step as independent, thus aligning with single process parallel machine shop manufacturing environment. However, majority of discrete production factory have flow shop or job shop manufacturing environment.

Most of the existing research on energy efficient scheduling focuses on flow shop modeling. However, flow shop is a special case of job shop. Job shops are much complex as compared to flow shops because of the flexibility in job routing and job sequencing. Job shop scheduling is a classic example of NP - Hard problems. The major objectives of this Chapter include:

- Developing an intelligent algorithm for dynamic job shop scheduling
- Integrating production line loads with non-production line loads and on-site energy generation in a multi-objective job shop scheduling problem.
- Using forecasting as input in the dynamic load scheduling problem

The model developed in this chapter is different from the models developed in previous chapter as:

- The algorithm is based on local controller's perspective as against a Central Operator in the previous Chapter
- Energy forecast for future time slots is used for making scheduling decisions at every time slot t

In Demand Side Management (DSM), demand requests are generated by local controllers (e.g. smart meters installed at consumers physical location), a central μG controller gathers real-time information, develops optimal control commands and sends them to the local controllers. The control commands can include different measures e.g. maximum peak or maximum energy allowed to be consumed in the next time slot and the consumer will be charged based on their actual consumption. If the actual consumption by consumer is higher from the control command sent by the μG central controller, the consumer is faced with higher electricity prices [260]. In this Chapter, an intelligent algorithm is developed for a job shop manufacturing facility that can generate dynamic demand requests to the central μG controller.

Many real-time and online DSM algorithms have been developed for smart grid as some of them are highlighted in Table 4.1. The research articles can be classified into two different types. 1) Type I: from a central controller perspective and 2) Type II: from a local controller perspective. In Type I, the models obtain optimal command controls to be sent to the local controllers while in Type II, the model obtains optimal demand requests to be sent to the central controller. In Type I, the model assumes that the demand requests from consumers is optimized from consumers perspective. Shi et.al ([208]) proposed a research direction to develop intelligent algorithms on the consumer side to generate the demand requests. While most of the existing works focus on residential or commercial consumers where non-critical loads can be reduced or shifted conveniently [261], an industrial consumer is considered in this Chapter with job shop manufacturing set-up to generate demand requests for the μG central controller.

4.1 Job shop scheduling

Unlike the various works in DSM, energy efficient job shop scheduling is strictly a multi-objective problem. This is because any production scheduling problem has a traditional objective e.g. makespan, tardiness or throughput that is a measure of

Table 4.1.
Classification of research articles based on load scheduling perspectives

Article	Type	Method	Consumer type
[24]	I	Game Theory	Residential
[262]	I	Game Theory	Not specified
[206]	I	Lyapunov Optimization	μGs
[263]	I	Interval Optimization	Not specified
[208]	I	Lyapunov Optimization	Not specified
[51]	II	Game Theory	Residential
[260]	II	Welfare maximization	Not specified
[209]	II	Game Theory	Residential
[148]	II	Integer Programming	Flow shop
[261]	II	Scenario generation	Industrial

performance of the scheduling algorithm. Job shop scheduling with energy efficiency makes the problem multi-objective. In this Chapter, three objectives are considered 1) minimizing total electricity cost and 2) minimizing average tardiness of jobs 3) minimizing average peak consumption.

Job shop scheduling is NP hard in nature. In a job shop manufacturing facility, an order needs to go through multiple machines before it becomes a final product. In flexible job shop, an task or operation of an order can be processed in multiple candidate machines. Different characteristics of job shop scheduling are:

- A single order/operation/task can be processed in a machine at a given time. The number of units being processed depends on the batch size
- No preemption is allowed, that is, if an order starts processing in a machine, it cannot be removed until it is finished

- Processing speed must be selected before starting an order in a machine. Once the order starts processing, the processing speed cannot be changed.

- The precedence constraints have to be strictly followed and operation sequence of an order follows a Directed acyclic graph

- A machine once started, should run for a minimum of some time to avoid frequent switching on/off of machines

- When a machine is started, there is an instantaneous spike in the power consumption

In this Chapter, an intelligent algorithm is proposed for the job shop facility through a local controller. In the previous chapter, all the information was shared with the central controller. using this algorithm, the individual consumer privacy can be maintained and only request messages are sent from the consumers to the central controller. The schematic representation is shown in Figure 4.1. Shi et.al [208] proposed a research direction to develop intelligent algorithms on the consumer side to generate the demand requests.

4.2 Problem Set-up

In this Chapter, the focus is on the production line loads of an industrial consumer and not the μG with all the other consumers. The job shop scheduling aims at minimizing the time average electricity cost over infinite horizon such that a certain due date criteria is met. Let the job shop has $|J|, j = 1, 2, ..., J$ machines, it can process $|K|, k = 1, 2, ..., K$ types of jobs and receives orders $O, o = 1, 2, ..., |O|$. Each order has a due date dd_o. Each job type k has set of tasks $V_k, v = 1, 2, ..., |V_k|$ to be completed where the known and deterministic processing time and power consumption of each task v is given by $T^o_{k,v}$ and $P^o_{k,v}$ respectively. Task v for job type k is performed in machine $m(k, v)$. A machine j consumes power IM_j when sitting idle, OM_j when it is switched on or OF_j when it is switched off where OM_j and OF_j are

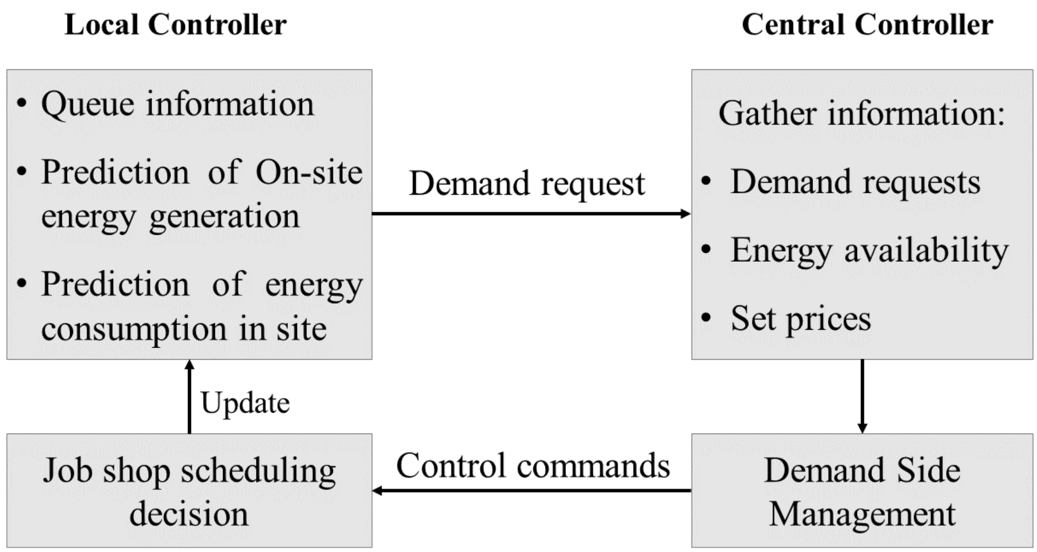


Fig. 4.1. Schematic representation of the proposed algorithm

instantaneous. These factors make the production loads significantly different from building loads.

In dynamic production scheduling, heuristics such as First Come First Serve (FCFS), Shortest Processing Time (SPT), Earliest Due Date (EDD) or Critical Ratio (CR) are employed in manufacturing facility based on the objectives of the facility. CR is the best performing heuristics while considering minimizing tardiness. In CR heuristics, orders in a machine are prioritized based on the value of $CR_o = \frac{dd_o - t}{RPT_o}$, where RPT_o is the processing time remaining for completion of order o. In CR heuristics, the lower the value, higher is the priority. Let the service rate of a task is $\mu_{j,k}$ that is the inverse of its processing time. $\mu_{k,v} = 0$ if $j \neq m(k,v)$.

The model is based on certain assumptions. The assumptions considered in this Chapter are:

- the due date of an order o is given $dd_o = t + RPT_o + u\text{x}RPT_o$ where u is a random numer drawn from uniform distributionũniform$(0,1)$.

- the model is discrete time with 15 minutes time slots and industrial consumer has on-site energy generation
- energy generation from renewables follow a stationary time series
- arrival process of order for type k follows Bernoulli process with nominal rate $0 < \lambda_k < 1$, that is, jobs of type k arrive in every time slot with probability λ_k.
- the industrial consumer wants to schedule jobs such that the peak is below a predetermined level P.
- the proposed intelligent algorithm predicts the future values of on-site generation and $x^{NS}[t]$ for a fixed T' future time slots.

On-site energy generation could harvest $H[t]$ at time t. We model the problem as infinite horizon problem as demands are updated in the system based on online orders from the customers. Prices for time t are assumed to be known. The production facility draws $G[t]$ from grid and $x^{NS}[t]$ is the non-production line load at time t. They can also sell $S[t]$ back to the grid at \$$p_4[t]$/KWh. The facility has a battery with capacity $A_{max}, Y_{max}, B_{max}$ for storage, charging and, discharging respectively while the respective amounts at time t are given by $A[t], Y[t]$ and $B[t]$.

The objective functions are shown in Equation 4.1, Equation 4.2 and Equation 4.3. Equation 4.1 is minimizing the time average cost. Equation 4.2 is minimizing the average tardiness where CT_o is the completion time, R_o is the arrival time and $FT_o = CT_o - R_o$ is the flow time of order o. Equation 4.3 is minimizing the peak minimization where $L[t]$ is explained in Equation 4.8.

$$min \qquad \lim_{|T|\to\infty} \frac{1}{|T|} \sum_{t\in T} G[t]p_1[t] + H[t]p_2[t] - S[t]p_4[t] \tag{4.1}$$

$$min \qquad \frac{1}{|O|} \sum_{o\in O} \left[CT_o - dd_o\right]^+ \tag{4.2}$$

$$min \qquad \left(max\ L[t] \qquad \forall t \in T \right) \tag{4.3}$$

The constraints for the battery are shown by Equation 4.4 - Equation 4.6. Evolution of battery is given in Equation 4.4. The capacity constraint are given by Equation 4.5. Capacity feasibility constraints are shown in Equation 4.6.

$$A[t+1] = A[t] + Y[t] - B[t] \qquad \forall t \in T \tag{4.4}$$

$$A[t] \leq A_{max}, B[t] \leq B_{max}, Y[t] \leq Y_{max} \qquad \forall t \in T \tag{4.5}$$

$$B[t] \leq A[t] \text{ and } Y[t] \leq A_{max} - A[t] \qquad \forall t \in T \tag{4.6}$$

Let $L[t]$ be the power load from the manufacturing site at time t. The supply demand matching is given in Equation 4.7.

$$G[t] + H[t] + B[t] = L[t] + Y[t] + S[t] \qquad \forall t \in T \tag{4.7}$$

$L[t]$ is the total power consumed in time slot t as given in Equation 4.8. It consists of machine operation (idle/on/off), processing of a task and non-production line load. At any time t, let there be Q^j orders queued in machine j and let $P^o_{j,q}$ is the processing power and $T^o_{j,q}$ is the processing time for q^{th} order in machine j. These values are known as we know the task and job type of each element in the queue as after each task, the order moves to the queue of the next task. Let $x_{j,q}[t] = 1$ if q^{th} order of machine j is running at time t. Let $ms_j[t] = 1$ if machine was running at time $t-1$, $y_j[t] = 1$ if machine was switched on at time t,$z_j[t] = 1$ if machine was switched off at time t and $v_j[t] = 1$ if machine is kept running idle at time t and $ML_j[t]$ is the machine load as explained in Equation 4.10. The domain constraint is shown in Equation 4.11.

$$L[t] = x^{NS}[t] + \sum_{j=1}^{M} \left(\sum_{q=1}^{Q^j} x_{j,q}[t] + ML_j[t] \right) \qquad \forall t \in T \tag{4.8}$$

$$\begin{aligned} ML_j[t] =& OM_j y_j[t](1 - ms_j[t]) + IM_j v_j[t] ms_j[t] \\ &+ OF_j z_j[t] ms_j[t] \qquad \forall j \in J, \forall t \in T \end{aligned} \tag{4.9}$$

The constraints for running status of a machine are given in Equation 4.10.

$$
\begin{aligned}
v_j[t] + z_j[t] \leq 1 \qquad & \text{if} \quad ms_j[t] = 1 \\
v_j[t], z_j[t] = 0 \qquad & \text{if} \quad ms_j[t] = 0
\end{aligned} \tag{4.10}
$$

$$
\begin{aligned}
v_j[t], y_j[t], z_j[t], x_{j,q}[t] \in \{0, 1\} \\
L[t], X[t], S[t], H[t], Y[t], B[t] \in R^{+}
\end{aligned} \tag{4.11}
$$

Apart from these constraints, some of the constraints are explained in Section 4.2.1. Constraints on $L[t]$ are further explained in Section 4.3.1 which are essential to maintain the job shop assumptions. Before that, the capacity region of the scheduling is derived next.

4.2.1 Capacity Region for Scheduling

Capacity region (Cap) is used to define the processing capability of the job shop. It ensures that the queues remain stable for the given arrival rates. Let machine j allocates $\rho_{j,k,v} \leq 1$ proportion of its resources to task v of job type k. We define a set $K' = 1, 2, ..., \sum_{k=1}^{N} K_v$ that contains the list of all the tasks. Let N^j be the set of feasible configurations for machine j. The capacity region for λ_k is defined by solving the following Linear Programming (LP) problem.

$$
max \qquad \sum_{k=1}^{N} \lambda_k \in Cap \tag{4.12}
$$

such that

$$
\lambda_k \leq \left(\mu_{j,k,v} \rho_{j,k,v} \ \forall v \in V_k, \forall j \in J \right) \qquad \forall k \in K \tag{4.13}
$$

$$
\sum_{k=1}^{N} \sum_{v=1}^{V_k} \rho_{j,k,v} \leq 1 \qquad \forall j \in J \tag{4.14}
$$

Equation 4.12 aims at maximizing the capacity region of the job shop. Equation 4.13 ensures that the capacity is defined based on the slowest processing step, where right hand of the equation is the measure of effective rate of service for a task. Equation 4.14 is the capacity constraint for resource allocation.

Let $V = \cap_{k=1}^{|K|} V_k$ be a vector of all the different tasks from all the different job types to be scheduled in the facility. Let $e_{\bar{v}}$ be a vector with all 0 vectors except 1 at location $\bar{v}$. Let $N^j = (e_{\bar{v}} \, \forall v \in V_k, \, \forall k \in K)$ if $j = m(k, v)$. Similarly we define P^o as vector for power consumption by the different tasks.

$$\lambda_k \in \bar{\bar{N}} = \left\{ Conv(N^j) \qquad \forall j \in J, \forall k \in K \right\} \tag{4.15}$$

$$\lambda_k \in \bar{\bar{N}} \Bigg| \left(\sum_{j \in M} \sum_{v \in \bar{V}}^{K} e_v P_v^o < cm - x^{NS} \right) \tag{4.16}$$

Let the minimum value of control command from central μG be cm and x^{NS} be the maximum non-production line load. Equation 4.15 provides the feasibility constraint when $Conv(N^j)$ is the convex hull. Equation 4.15 shows job shop specific constraint that a machine can process only one job at a time and any task of a job can be processed in a single machine only. Equation 4.16 provides the feasibility constraint for maximum peak load for production line load. Strict inequality is used to account for machine loads in Equation 4.9.

Lemma 1 $\frac{1}{|O|} \sum_{o=1}^{|O|} \left(CT_o - dd_o \right) \leq 0$.

Proof Let the completion rate of an order be de_o such that $\frac{1}{FT_o} = de_o$ and $FT_o \geq RPT_o$. Since $\lambda_k \in C$, $\lambda_k \leq de_o$ (arrival rate less than service rate for queue stability), $FT_o \leq \lambda_k^{-1}$. Let jobs of type k arrive with arrival rate λ'_k and $FT'_o = dd_o - R_o \implies RPT'_o \geq RPT_o$. For such jobs, at least one of $\mu'_{j,k,v} \leq \mu_{j,k,v}$ as $RPT'_o \geq RPT_o$. Since $\lambda'_k \leq \mu'_{j,k,v} \rho_{j,k,v}$ and thus $\lambda'_k \leq \lambda_k$ and $\lambda'_k \in C$. $\lambda'_k \leq \lambda_k$ implies $FT_o \leq (\lambda'_k)^{-1} \leq FT'_o$ which shows that $CT_o - dd_o \leq 0$ where $CT_o \leq FT_o + R_o$. Summing this result over all $o \in O$ completes the proof. In essence, this lemma shows that since the arrival rate of jobs is within the capacity region, the queues in machine do not blow up, hence the tardiness does not blow up.

4.2.2 Virtual Queues for Lyapunov Optimization

The optimal policy for minimizing tardiness is to follow CR heuristics. Thus objective function in Equation 4.2 is minimized if we follow a CR policy. However, it is not optimal for the other 2 objectives. We devise a control policy that attempts to follow the scheduling policy close to CR but also considers other objective functions. To do this, we propose two virtual queues shown in Equation 4.20 - Equation 4.21 where $\delta \geq 0$ and $\chi \geq 0$ are two control hyper-parameters for the managers.

Let the orders in machine j queue are denoted by $Q^j[t]$. We assume that the orders are placed in a queue based on their CR values. $x_{j,q,t} = 1$ if the proposed policy schedules the order in sequence q at time t. Let the q^{th} order in the queue of machine j is has priority $pr_{j,q}$, processing power $P^o_{j,q}$ and processing time $T^o_{j,q}$. T' is the maximum orders that can be scheduled in the next T' time slots.

$$
\begin{aligned}
pr_{j,q} &= |Q^j[t]| - q + 1 \Bigg| \sum_{q=1}^{|Q^j[t]|} T^o_{j,q} \leq T' \\
pr_{j,q} &= |Q^j[t]| - q + 1 \Bigg| \sum_{q=1}^{q'-1} T^o_{i,q} < T' \leq \sum_{q=1}^{q'} T^o_{j,q}, |Q^j[t]| \leq T' \\
pr_{j,q} &= [T' - q + 1]^+ \Bigg| |Q^j[t]| \geq T'
\end{aligned}
\tag{4.17}
$$

$arr[t]$ and $dep[t]$ is defined in Equation 4.18 and Equation 4.19 respectively. $pr_j[t]$ is given by Equation 4.17. The priority value is defined as such to make $arr[t]$ bounded and also it balances the workload amongst the different machines by not giving excessive weight to a machine with higher queue length. In this Chapter, future prediction up to time $t + T'$ is used for making decision at time t. Intuitively, $arr[t]$ is the sum of priorities of all the jobs that could be started in the time slots $t' = \{t, t+1, ..., t+T'\}$ while $dep[t]$ is the sum of priority of jobs that the proposed policy schedules in the time slots t'.

$$
arr[t] = \sum_{j \in M} \sum_{q=1}^{|Q^j[t]|} pr_{j,q}[t] \tag{4.18}
$$

$$dep[t] = \sum_{i=1}^{N} \Bigg(\sum_{q=1}^{|Q^j[t]|} p_{j,q} x_{j,q} \Bigg) \tag{4.19}$$

$$Q_d[t+1] = \big[Q_d[t] + arr[t] - dep[t] - \delta\big]^+ \qquad \forall t \in T \tag{4.20}$$

$$Q_p[t+1] = \big[Q_p[t] + L[t] - (P - \chi)\big]^+ \qquad \forall t \in T \tag{4.21}$$

Objectives in Equation 4.2 and Equation 4.3 can be converted to a time averaged constraint as shown in Equation 4.22 and Equation 4.23 respectively. The two virtual queues above are derived from these objective turned constraints.

$$\lim_{|T| \to \infty} \frac{1}{|T|} \sum_{t \in T} \big(arr[t] - dep[t]\big) \leq \delta \tag{4.22}$$

$$\lim_{|T| \to \infty} \frac{1}{|T|} \sum_{t \in T} L[t] \leq P - \chi \tag{4.23}$$

It can be seen from the Equation4.20 and Equation 4.21 that δ and χ are the hyper-parameters acting as two levers that can be controlled by the managers at the industrial facility. The effect of changing δ and χ is shown in Section 4.4.

Lemma 2 $arr[t] \geq dep[t]$ and $arr[t] < \infty$

Proof It can be seen from the definition of $arr[t]$ and $dep[t]$ that $arr[t] \geq dep[t]$ since orders are arranged according to CR priority while maximum jobs possible are scheduled. The priority of an order is bounded by T', thus $arr[t] < \infty$

Lemma 3 CR policy is optimal for objective function given in Equation 4.22 and Equation 4.2.

Proof CR policy is the optimal policy for the objective of minimizing tardiness in Equation 4.2. Assume $E[Q_d[0]] = 0$. If we follow the CR policy, $arr[t] = dep[t]$, $\forall t \in T$. Thus CR policy ensures the feasibility of Equation 4.22 even when $\delta = 0$. Since, we are not following a CR policy and devising our own policy, δ is required to make the virtual queue Q_d stable.

Theorem 1 If $\frac{1}{|T|}\sum_{t\in T} E\big[arr[t]-dep[t]\big] \leq \delta$, queue $Q_d[t]$ is mean rate stable.

Proof Consider Equation 4.20, it can be observed that $Q_d[t+1] \leq Q_d[t] + arr[t] - dep[t] - \delta$. Taking the expectation on both sides and summing it over all t, gives Equation 4.24.

$$\begin{aligned} E[Q_d[T]] - E[Q_d[0]] &\leq \sum_{t\in T} E\big[arr[t]-dep[t]-\delta\big] \\ \frac{1}{|T|}E\big[Q_d[T]\big] &\leq \frac{1}{|T|}\Bigg(\sum_{t\in T} E\big[arr[t]-dep[t]\big]\Bigg) - \delta \end{aligned} \tag{4.24}$$

If the condition proposed in this theorem is satisfied, $\frac{1}{|T|}E[Q_d[T] \leq 0$, making it mean rate stable. Similarly, it can be shown that since $E\big[L[t]\big] \leq P - \chi$, $Q_p[t]$ is mean rate stable.

δ plays an important role in making the virtual queue stable and also increasing the value of virtual queues that would create pressure on the Lyapunov iteration step to schedule jobs with higher CR priority. Similarly, χ is a measure of safety in terms of peak maintenance. Lower the value of χ, higher is the value of the virtual queue thus there is more pressure to schedule jobs with lower power consumption.

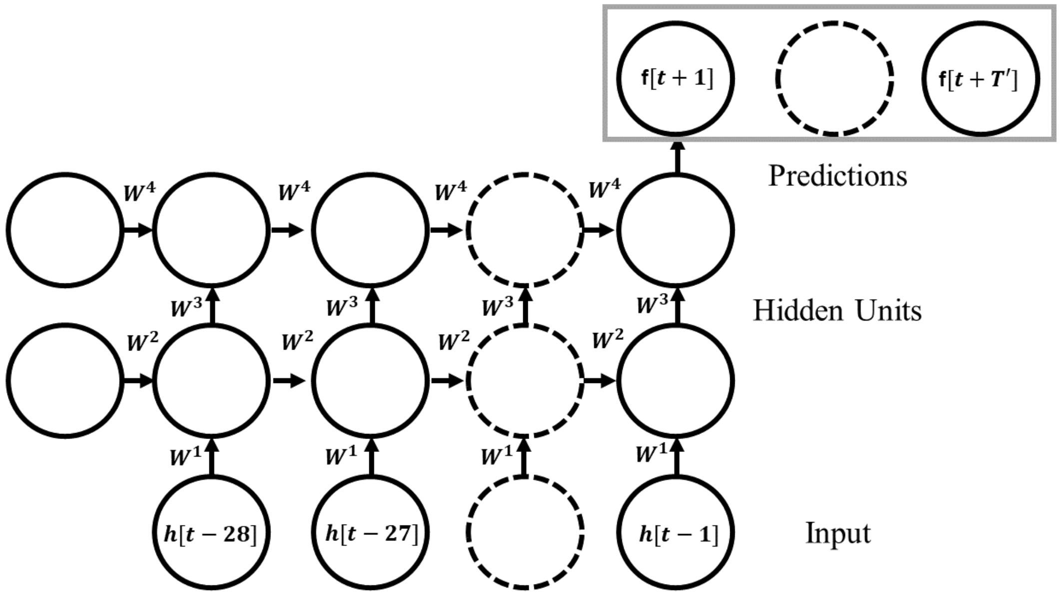
4.2.3 Predicting on-site Energy Generation

In this research, energy from on-site generation is predicted using a ensemble method involving a Recurrent Neural Network (RNN) and ARMA (Auto-Regressive Moving Average) time series model. In this Chapter, a wind mill is considered that is installed at the manufacturing facility that supplies renewable energy to the plant. Similar analysis and study can be conducted for a facility with solar energy on site generation. However, the characteristics will be different from the wind energy generation model developed in this Chapter. Please refer to [264] for understanding the recurrent neural networks. The data for training the model is obtained from ERCOT where power form the wind mill is recorded every hour [252]. It is assumed that f_t follows a multivariate normal distribution $N(f_t, \sum_f)$ where f_t are the forecast made at t for $t' \in \bar{T} = \{t+1, t+2, ...t+T'\}$.

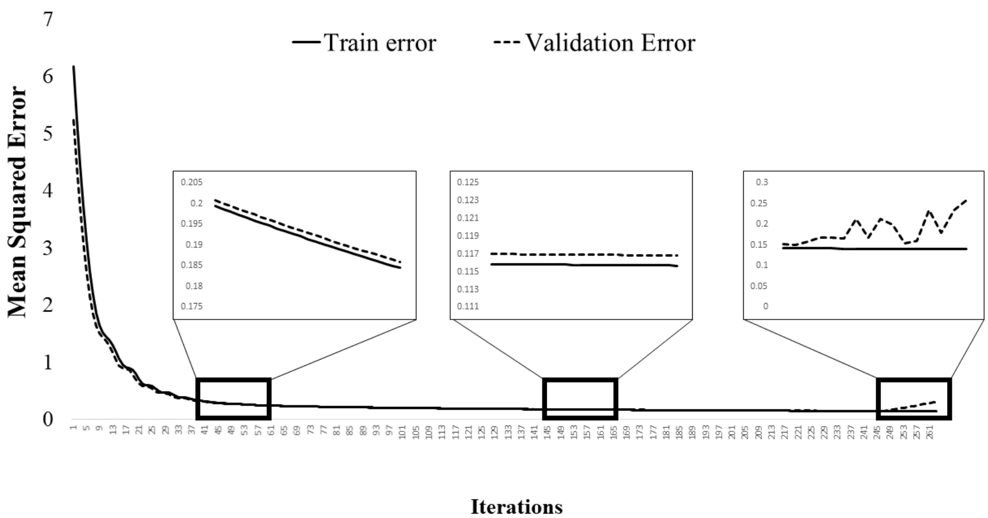
The method consists of using the average of two predictions. One vector is predicted using recurrent neural network and one vector is predicted using time series model. And the arithmetic mean of the two vectors is used as f_t. The best model for prediction in this research is obtained using training the RNN based on the square loss function. The architecture of the RNN and decreasing loss function with training iterations is shown in Figure 4.2(a) and Figure 4.2(b). Figure 4.3 shows a sample prediction against true value using RNN model. In Figure 4.3 and Figure 4.5, the blue line shows the true values while the green line shows the prediction. The loss functions shows that the RNN model has converged. The training stops when the out - of - sample error, or, errors in test validation data starts increasing. For RNN, a two layer perceptron architecture is considered. The model considered time lags of 28 time slots to predict f_t.

A RNN architecture is preferred over multi-layer perceptron model as multi layer perceptron model does not consider the dependency between the data input. In the case of predicting power from wind, the future predictions are heavily correlated with the past data. A RNN architecture is the state of the art architecture for predicting time dependent values. The RNN architecture considered in this research uses Long Short Term Memory (LSTM) cells. Multilayer perceptron model and RNN model in particular face the problem of vanishing gradient during the updating (using back propagation) of the weights. LSTM cells in the RNN architecture handles the problem of vanishing gradients that is prevalent in RNN. A learning rate of 0.05 is considered for training the model and the batch size is kept at 32. The number of hidden layers is set at 2 as the model performance increases when increasing the number of layers.

The performance of RNN model is better as T' increases. However, when predicting for near future time slots, the performance is not as good because it does not handle the correlation between the time slots mathematically and independently of other time slots. Thus, past plays a major role in predicting near future time slots and acts in contributing noise to the estimation. However, the time series model mathematically handles the past that should be included in the model to predict the



(a) Architecture for Recurrent Neural Network



(b) Model training error with iterations for RNN

Fig. 4.2. RNN model for estimating f_t

future time slots. Therefore a time series model is also studied to predict the power available from the wind mill to the facility. However, the prediction performance deteriorates as T' increases because of the following two reasons:

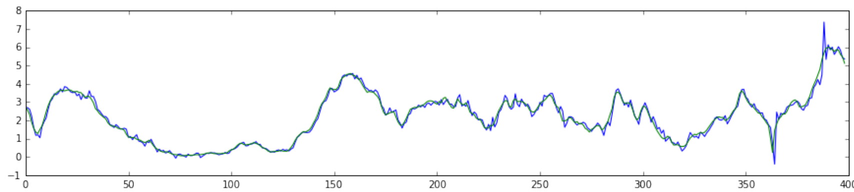


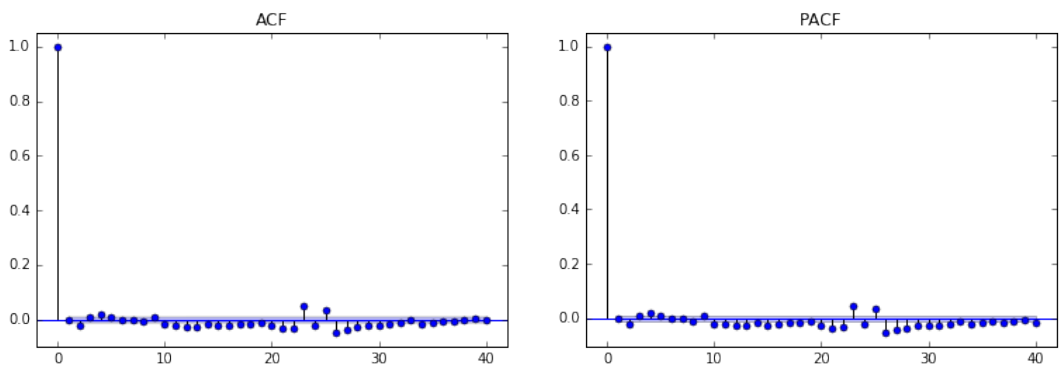
Fig. 4.3. Prediction sample for time series model

- the time series model takes time to recover from shock. When the time series model experiences a randomly high/low value, it takes time to wash off the residue as the prediction is based on this randomly high/low value
- the prediction of the time series model decays exponentially with increasing future time slot. Thus the performance of the model decreases for increasing future time slots

Because of these two reasons, time series model is used with RNN model for power prediction from wind mill. Thus, the two models have their advantages and disadvantages and an ensemble methods performs better than the two methods considered individually. Next, the time series considered in this research is explained.

For ARMA time series model, a ARMA (3,0,0)x (1,0,0,24) model is fit. ARMA model provides better predictions for smaller lookahead time slots but the prediction performance decreases as the prediction time horizon increases. If w_{t+1} is the prediction for time slot $t+1$, the function for prediction using ARMA is shown in Equation 4.25. The coefficients are obtained for the simplest model with the best fitting function. The best fit is considered based on root mean square values and Akaike Information Criteria (AIC) values. The Autocorrelation function (ACF) and the partial autocorrelation function (PACF) of the residuals is shown in Figure 4.4(a) and Figure 4.4(b) respectively. A sample of prediction from time series model is shown in Figure 4.5. The ACF and PACF plots show that the model fit is good and the residuals could represent white noise (mean 0 and a constant variance σ).

$$
\begin{aligned}
w_{t+1} = & 1.6478 w_t - 0.7428 w_{t-1} + 0.08 w_{t-2} + 0.1523 w_{t-24} - && (4.25) \\
& 1.6478 * 0.1523 x_{t-24} + 0.7428 * 0.1523 x_{t-25} - 0.008 * 0.1523 x_{t-26} && (4.26)
\end{aligned}
$$



(a) ACF of the time series model (b) PACF of the time series model

Fig. 4.4. Time series model for estimating f_t

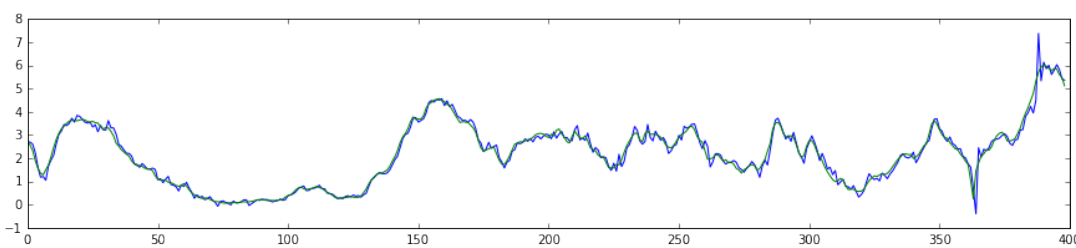


Fig. 4.5. Prediction sample for time series model

4.3 Handling uncertainty in power prediction

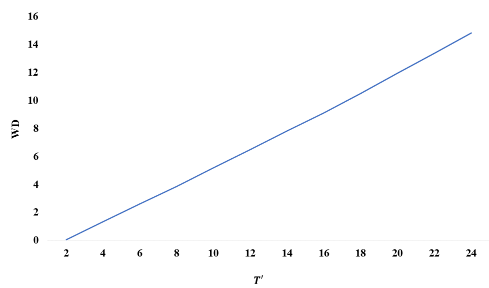
Power from the wind mill is highly unpredictable because of the uncertain nature of the weather. Thus estimating the power from just one data point (based on recent history) might lead to highly noisy and poor predictions. Thus, when predicting on-site energy generation, f_t is obtained by sampling. Sample size S is decided based on Wasserstein Distance (WD) as explained By Bao et. al [261]. If the prediction horizon T' is smaller, prediction accuracy is higher as against when T' is large as shown by increasing TD and T' in Figure 4.6(a). Let $\bar{S}$ be the set of samples obtained for the prediction. Let the prediction follow a multivariate distribution with parameters $N(f_T, \sigma_{f_t})$. From the dataset, different data points are collected for prediction at point t such that the data points were obtained for time of the day in terms of hour t. The heuristic used to draw samples is described in Heuristic 4.1. Future values of these data points are considered as true values and the WD distance is calculated using these values. Let the true values follow a distribution given by a multivariate distribution with parameters $N(\bar{f}_T, \sigma_{\bar{f}_t})$. Let the correlation coefficients for the 2 distribution is given by C_f and $\bar{C}_f$. The WD distance between these two distributions is given as shown in Equation 4.27.

$$WD = |f_t - \bar{f}_t| + Trace\left(C_f + \bar{C}_f - 2\sqrt{\sqrt{C_f}\bar{C}_f\sqrt{C_f}}\right) \tag{4.27}$$

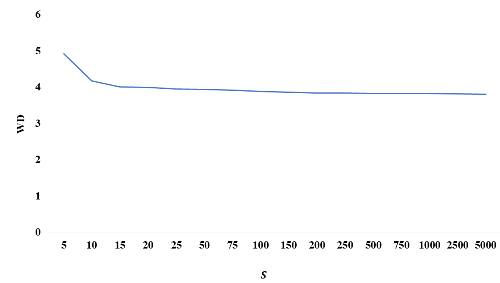
Heuristic 4.1: Drawing samples for prediction

1. For every time slot h, collect the values of previous T' time slots in set F_t
2. Calculate the exponential average of the elements in the set F_t
3. Run a K-means clustering algorithm with $K_c = 4$ for separating the data in each time slot into 4 different groups and calculate the average of each group
4. When drawing samples for hour h, calculate the exponential average ea_h of its previously observed T' time slots and find the cluster with its average closest to ea_h
5. Draw samples $\bar{S}$ randomly from this cluster for a given hour of the day h

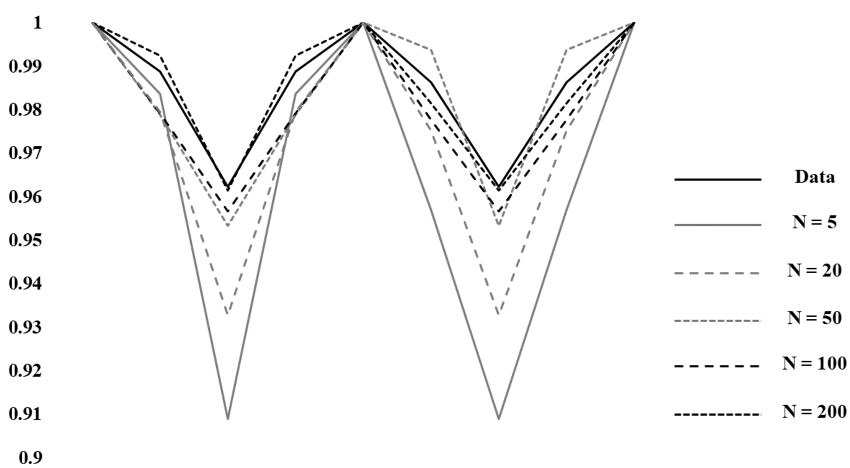
As $\bar{S}$ increases, WD decreases when measured against true data. In this Chapter, 200 samples are drawn for estimating f_t as there is no further improvement in WD on increasing S. An example of correlations coefficients is shown for $T' = 3$ in Figure 4.6(c) which shows that increasing sample size improves estimation. The correlation matrix is converted into a single vector and drawn as shown in Figure 4.6(c). The x axis are the numbers from 1 to 9 representing the index of the correlation values. The fitted distribution is assumed to be a multivariate distribution with T' dimensions.



(a) WD with T' for $S = 200$



(b) WD with S for $T' = 6$



(c) Correlation with increasing S

Fig. 4.6. Wasserstein distance with different T' and S

4.3.1 Lyapunov Optimization Iteration Step

Based on the previous Chapters, Lyapunov Optimization aims at minimizing drift + penalty function at each time step to achieve different objectives. For a Lyapunov iteration step, the future values for electricity generation from renewables and non-production line load are considered. The non-production line load $x^{NS}[t']$ is estimated as the average for that time slot for all the previous days. $f_t = \{f[t+1], f[t+2], ..., f[t+T']\}$ is obtained by prediction explained in Section 4.2.3.

Define $\Theta[t] = (Q_d[t], Q_p[t])$. The Lyapunov function and one slot conditional drift for virtual queues $Q_d[t]$ and $Q_p[t]$ is given by Equation 4.28 and Equation 4.29 respectively, where the value of B is given in Equation 4.30. α_d and α_p are the weights given to the two virtual queues and B is obtained from the definition of $arr[t]$ that sets the maximum value of priority to be T'. Since a maximum of T' orders can be schedule in a machine in T' time slots, fist half of B is obtained by the sum of the maximum possible priorities.

$$L(\Theta[t]) = \frac{1}{2}\Big(\alpha_d Q_d^2[t] + \alpha_p Q_p^2[t]\Big) \tag{4.28}$$

$$\begin{aligned} \triangledown(\Theta[t]) &= E\big[L(\Theta[t+1]) - L(\theta[t])|\Theta[t]\big] \\ &\leq \alpha_d Q_d[t] E\big(arr[t] - dep[t] - \delta|\Theta[t]\big) \\ &+ \alpha_p Q_p[t] E\big(L[t] - (P - \chi)|\Theta[t]\big) \end{aligned} \tag{4.29}$$

$$B = \frac{1}{2}\Bigg(\alpha_p M\Bigg(\frac{T'(T'+1)}{2} - \delta\Bigg)^2 + \alpha_d\big(P - \chi\big)^2\Bigg) \tag{4.30}$$

The upper bound on the weighted sum of drift-plus-penalty function,$\triangledown(\Theta[t]) + V\big(G[t]c_1[t] + H[t]c_2[t]\big)$ is minimized in each time slot t where V is another constant control variable. The $O(V)$ and $O(1/V)$ trade offs for drift and penalty can be contained for the problem considered in this paper, but the proof is similar as developed by Neely [201].

Thus, the job shop scheduling framework based on the virtual queues aims at solving the following optimization problem from Equation 4.31 - Equation 4.44 $\forall t \in T$. This Mixed Integer Programming Quadratic Constrained optimization problem is the intelligent algorithm at the industrial consumer's end. The objective function and most of the constraints are linear. The problem is quadratically constrained to handle the constraint in the manufacturing scheduling to ensure that once an order is started, it is not pre-emptied before completion. The storage constraints are satisfied in every time slot t. Power consumption for machine on/off/idle decisions and constraints for running an order makes the problem complex. These constraint differentiate the production line load from residential or commercial building loads.

$$
\begin{aligned}
& min \; -\alpha_d Q_d[t]dep[t] + \alpha_p Q_p[t]P + \mu P \\
& \quad L[t] + V\big(G[t]c_1[t] + H[t]c_2[t] - G[t]\big)
\end{aligned} \tag{4.31}
$$

such that

$$G[t] + H[t] + B[t] = Y[t] + L[t] \tag{4.32}$$

$$G[t'] + f[t'] + B[t'] = Y[t'] + L[t'] \; \forall t' \in \bar{T} \tag{4.33}$$

$$L[t] = x^{NS}[t] + \sum_{i=1}^{M} \sum_{q=1}^{Q^i[t]} P^o_{i,q} x_{i,q,t} + \sum_{i=1}^{M} ML_i[t] \tag{4.34}$$

$$L[t'] = x^{NS}[t'] + \sum_{i=1}^{M} \sum_{q=1}^{Q^i[t']} P^o_{i,q} x_{i,q,t'} \; \forall t' \in \bar{T} \tag{4.35}$$

$$L[t'] \leq P \qquad \forall t' \in \{t, ..., t+T'\} \tag{4.36}$$

$$\sum_{t'=0}^{T'} x_{j,q,t+t'} \leq T^o_{j,q} \qquad \forall q \in Q^j[t], \forall j \in J \tag{4.37}$$

$$x_{j,q,t+1}\left(\sum_{t'=t}^{t+T'-1} x_{j,q,t'}\right) \leq T^o_{j,q}, \;\; \forall q \in Q^j[t], \forall j \in J \tag{4.38}$$

$$x_{j,q,t'} \leq v_j[t] + y_j[t] \qquad \forall t' \in \{t, ..., t+T'\} \tag{4.39}$$

$$\sum_{q=1}^{Q^j[t]} x_{j,q,t'} \leq 1 \qquad \forall j \in J, \forall t' \in \{t, ..., t+T'\} \tag{4.40}$$

$$A[t'+1] = A[t'] + Y[t'] - B[t'] \, \forall t' \in \{t, ..., t+T'\} \tag{4.41}$$

$$A[t'] \leq A_{max}, B[t'] \leq B_{max}, Y[t'] \leq Y_{max} \, \forall t' \in \{t, ..., t+T'\} \tag{4.42}$$

$$B[t'] \leq A[t'] \text{ and } Y[t'] \leq A_{max} - A[t'] \, \forall t' \in \{t, ..., t+T'\} \tag{4.43}$$

$$x_{j,q,t'}, v_j[t'], y_j[t'], z_j[t'] \in \{0,1\} \forall t' \in \{t, ..., t+T'\} \tag{4.44}$$

$$G[t'], H[t'], L[t'] \in R^{+} \forall t' \in \{t, ..., t+T'\} \tag{4.45}$$

Equation 4.31 is the opportunistic drift-plus-penalty function. In Lyapunov optimization, there is a lag of 1 time unit. It means that a penalty at time t is reflected in the actions of time step $t+1$ after the virtual queue update. Thus a penalty μP is introduced on peak in Equation 4.31. Equations [4.32 - 4.33] are the supply demand matching constraints. Equations [4.34 - 4.35] expresses the power consumed at different time slots where $ML_i[t]$ is explained in Equation 4.9. Equation 4.36 ensures that the peak constraint is not violated. Equations [4.37 - 4.38] are the job shop constraints for non-interruption and non pre emptiveness as a task has to be completed once started. Equation 4.40 ensures the job shop constraint that machine can process a single job at a given time slot. Equation 4.39 shows a task can be completed only in a machine that is in on state. Equation 4.41 - Equation 4.43 are the storage constraints. Equations 4.44 - Equation 4.45 are the domain constraint.

The quadratic constraint may have a high computational time to handle the complexity in the constraint. However, since most of the constraints are linear and the problem is convex, the optimizer used in this research is able to solve the problem in small computation time. The optimizer uses interior point method to solve the problem to optimality. The optimality gap is set as 1x10^{-5} and the iteration stops when the solution achieves the optimality gap criteria. It is important that the computational time is small as the problem is solved for dynamic (and real-time) sequential decision making. If the computation time is large, the model needs significant changes to account for the lag in decision making. A study is provided in the Section 4.4 to understand the computational time to solve the problem.

After solving the model discussed above, $Q_d[t], Q_p[t]$ and $ms_i[t], \forall i \in I$ are updated and the orders with $x_{i,q,t} = 1$ are scheduled and start in time slot t. If an order is scheduled at time t, the power consumption of that order is added to non-shift able non-production line loads in the future time slots.

4.4 Simulation Experiments and Results

An example of the data for on-site generation profile, non-production line load profile and electricity price for a month is shown in Figure 4.7. The experiment is run for 180 days where first 150 days are discarded in analysis as warm up period. A job shop with 5 machines that can handle 5 job types are considered for analysis. Let u be a random drawn from uniform distribution $U(0,1)$. The number of tasks, processing time for each task and processing power for each task for each job type k is given by $K_j = 1 + \lfloor u\text{x}4 \rfloor$, $T^o_{j,k} = 1 + \lfloor u\text{x}4 \rfloor$ and $P^o_{j,k} = 25 + u\text{x}55$ respectively. As seen from Figure 4.7 and observed in an industrial facility, the non-production line load remains fairly stationary. Frequent switching on/off of machines is not good for the health of the machine. Based on the information from an industrial consumer, it is assumed that if a machine is switched off at time t, $ms_j[t'] = 0\, \forall t' \in \{t, t+1, ..., t+5\}$, thus machine is not switched on for the next 3 hours.

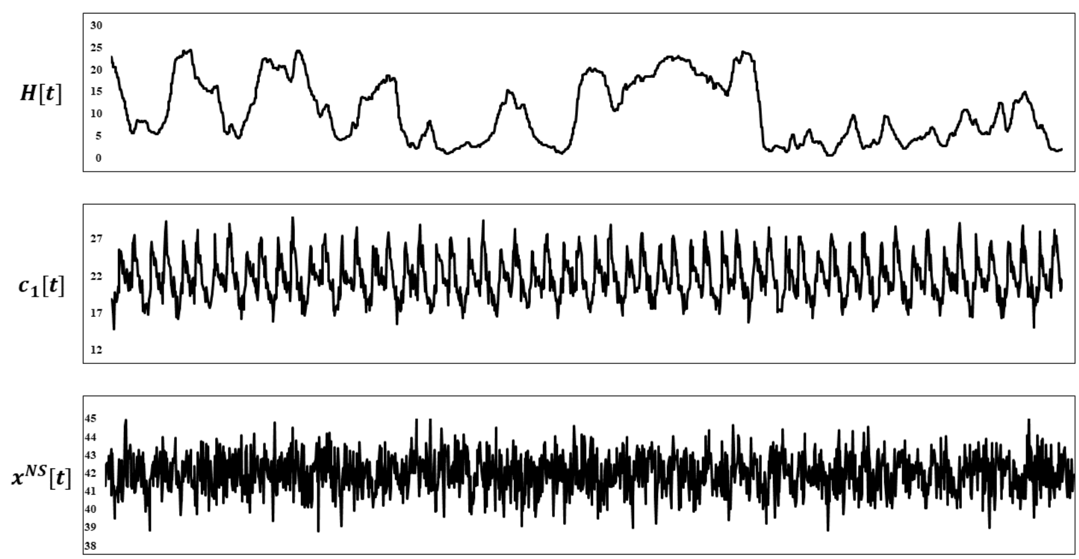


Fig. 4.7. Data used in this Chapter for simulation

Two different cases are considered. In Case I, the industrial consumers consider only previous day's demand, that is, at the start of the day, they have complete information of the orders to schedule for that day. In Case II, the demands are updated online in the system based on customers orders, thus there is higher stochasticity in demands. The former case is prevalent in industries under contract manufacturing while the latter case is prevalent in industries with emerging cloud manufacturing and Industry 4.0. All the experiment results shown in this Section show the average value when the experiment is run 30 times with different random number seed. Common random numbers are used to generate different experiments, thus maintaining the fairness in measuring the performance of the proposed framework.

Figure 4.8 - Figure 4.12 studies the affect of different hyper-parameters $\{\delta, \chi, T', V\}$. The sensitivity analysis is performed by changing one parameter while keeping other parameters as constant ($\alpha_d = 1, \alpha_p = 0.1, T' = 3, V = 25, \chi = 20, \delta = 3, \mu = 1, P = 0.8x|M|$x55).

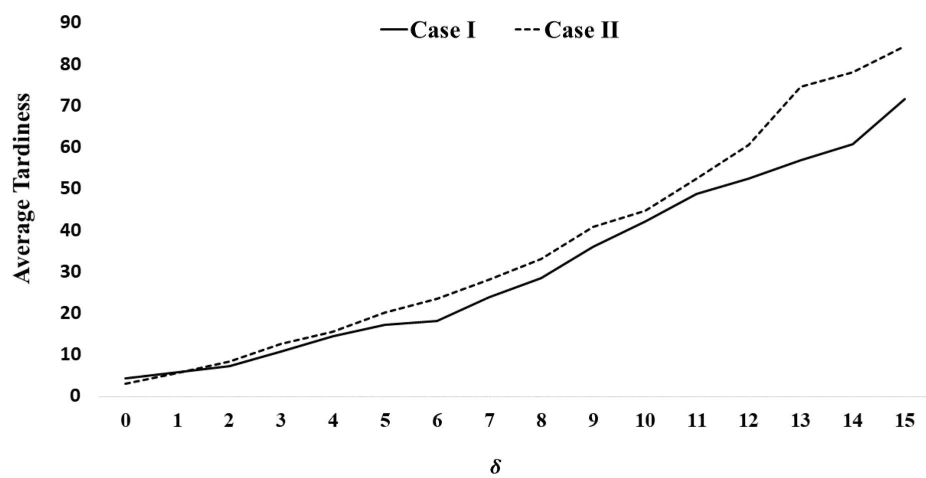


Fig. 4.8. Average tardiness with δ

As seen from the results in Figure 4.8, as the value of δ increases, the average tardiness increases as pressure on virtual queue $Q_d[t]$ decreases. Similarly, as the value of χ increases, average tardiness increases as shown in Figure 4.9. The pressure to reduce peak increases therefore fewer machines are switched on and fewer orders are

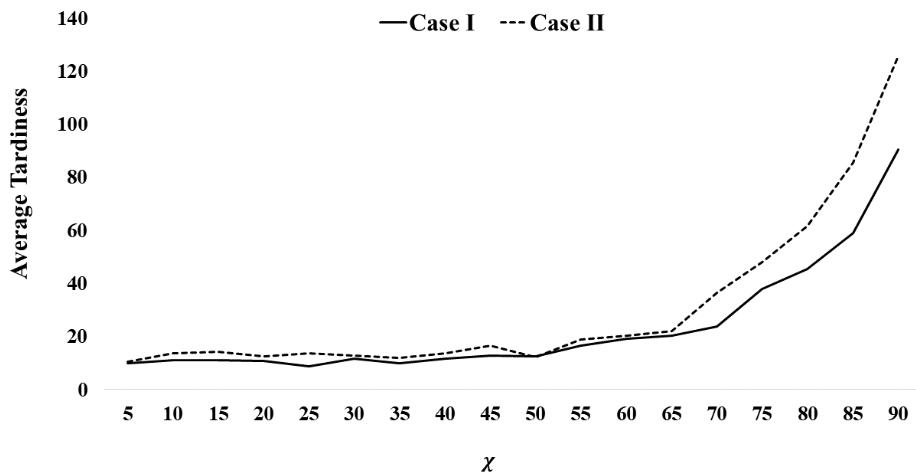


Fig. 4.9. Average tardiness with χ

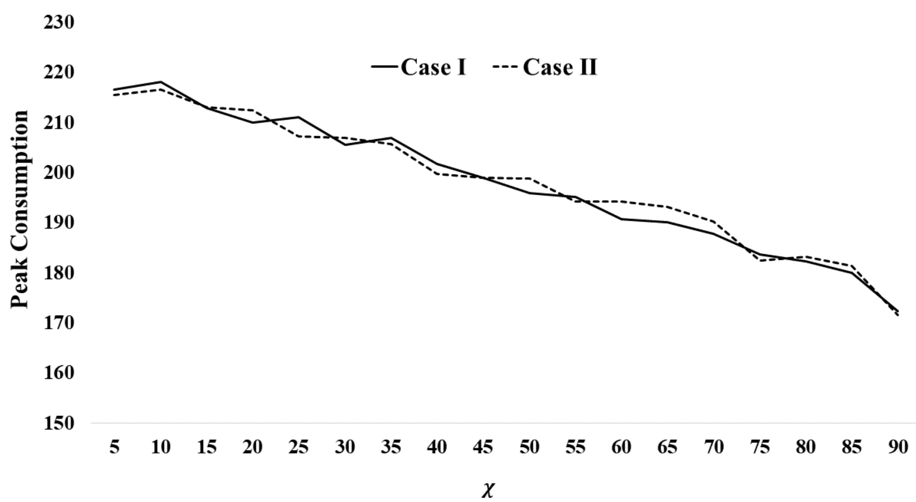


Fig. 4.10. Average Peak consumption with χ

scheduled. Also, as this pressure increases, the peak reduces as shown in Figure 4.10. Sensitivity for V is shown in Figure 4.11 in terms of the throughput percentage. This is because if the throughput is lower, electricity cost is lower as not enough jobs were scheduled for a fair comparison. As the value of T' increases, $Q_p[t]$ increases as

Fig. 4.11. Average throughput with χ

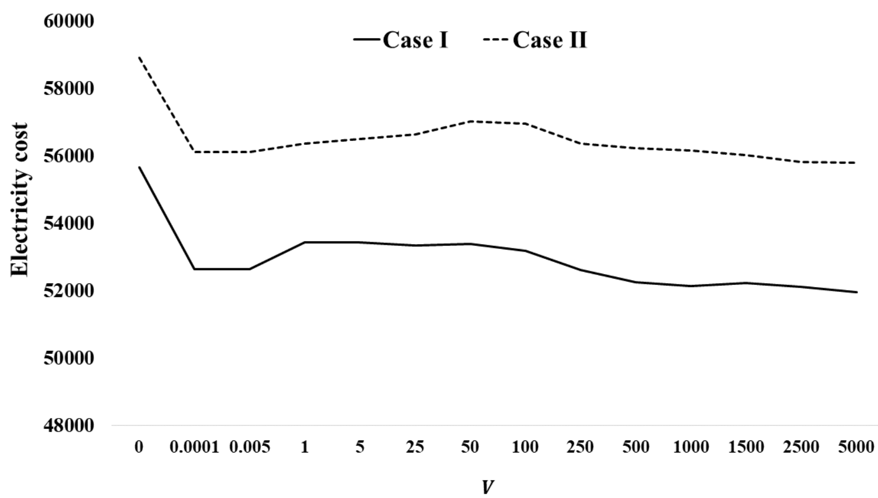


Fig. 4.12. Average electricity costs with V

more orders could be scheduled by CR but after a certain level, throughput starts decreasing as the orders are deferred to later t as number of orders in queue remains the same. It can be observed that $T' = 3$ performs the best as if T' is small, there is low pressure to schedule orders while the model has a tendency to defer orders when

T' is high. In this Chapter, $T' = 3$ is used as 6 half hour slots, thus, when prediction is used, 6 future time slots are considered. As V is increased, the cost first increases, then decreases. On increasing the V, the cost increases as drift term competes with penalty term in Equation 4.31, thus increasing the cost as more jobs are scheduled. However, as the value of V becomes large, drift term could only compete when it is too big, thus cost reduces as fewer jobs are scheduled.

This shows the importance of interplay between δ and χ in deciding the best hyper-parameters according to their metrics. The cost is highest when V is 0 as all the jobs are scheduled at earliest possible time.

The computational time for the Lyapunov optimization that is solved in every time slot is an important factor to enable real-time and dynamic decision making process. The average computation for solving the Lyapunov optimization step for different problem size is shown in Table 4.2. The smaller computational times ensures that the model can be applied in practice when making sequential decisions in every time slot. As observed in the table that as the problem size increases, the computational time increases as expected. A majority of small scale manufacturing facilities that could be placed in a small μG community have less than 30 machines. Large scale manufacturing facilities have 50 - 100 machines. Large scale manufacturing facilities may themselves be treated as independent μGs.

To evaluate the performance of the proposed framework, three different models are used: 1) $BM1$: CR ratio heuristics 2) $BM2$: Pure Lyapunov optimization without considering future expectations. For pure Lyapunov optimization, a max-weight policy is considered since $arr[t] = 0$ if $T' = 0$ is considered 3) $BM3$: A relaxed offline solution that relaxes some of the job shop constraints as job shop scheduling is NP-Hard and solves a deterministic problem. The problem is solved at the end of the period when all the stochastic variables values have been realized. The objective is to minimize the total electricity cost such that the throughput is $\geq$ the throughput of the proposed framework. To check the scalability of the model, different problem sizes are considered with higher values of M and N. The performance for the dif-

Table 4.2.
Computational time for different problem size

$(\lvert M\rvert, \lvert K\rvert)$	Time (in seconds)
(5,5)	0.083
(5,10)	0.092
(10,10)	0.279
(10,15)	0.318
(20,30)	0.469
(50,50)	0.832
(75,75)	1.491
(100,100)	2.936

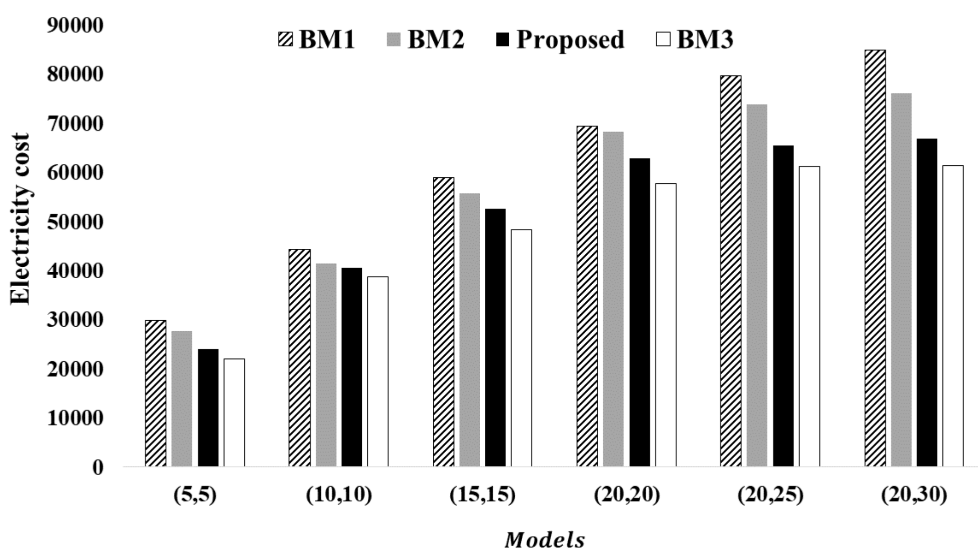


Fig. 4.13. Electricity cost for different models

ferent experiment set ups is provided in Figure 4.13. The results show that the CR heuristics performs the worst while the proposed heuristic performs comparative to the relaxed offline solutions obtained by solving the problem at the end of the period.

4.5 Conclusion and discussion

In this Chapter, an intelligent algorithm is developed for a local controller installed at consumer's location. A job shop manufacturing facility is considered that aims at minimizing the total electricity cost while reducing the average tardiness and peak energy consumption. The intelligent algorithm considers future expectations while making scheduling decisions at time t. This is the first work in dynamic job shop scheduling considering energy related objectives. The simulation results show that the algorithm performs comparatively to Critical Ratio (CR) heuristics in terms of tardiness while outperforming it significantly in terms of energy related objectives. The proposed framework in this Chapter introduces two hyper-parameters that can be used by the facility manager to control the throughput of orders and their effect is studied through sensitivity analysis. The manager can select the different values based on her/his preference for this multi-objective optimization problem.

The model can be extended in different directions. The value of T' is selected based on the sensitivity analysis. However, a theoretical study could be conducted to select the optimal T' based on the knowledge of the variance along with the sampling mean. The performance of the algorithm is compared with the benchmark solution obtained by solving the relaxed deterministic problem at the end of the planning period. This property is explored in Chapter 6 to learn a deterministic optimal policy based on the optimal actions by converting the load scheduling problem into a supervised learning with guided policy search. $X[t]$ in this Chapter can be considered as the request $r_i[t]$ from the industrial consumer to the SO. Similarly, intelligent algorithms can be built for different types of consumers. This Chapter considers an industrial consumer's perspective. In the next Chapter, central controller's perspective is considered where requests from different consumers can be developed using the intelligent algorithm proposed in this Chapter.

5. DYNAMIC LOAD SCHEDULING IN A μG WITH PARTIAL INFORMATION SHARING

In this Chapter, a dynamic load scheduling based on consumer utilities and partial information sharing is presented with different types of loads and different types of consumers described in a μG as discussed in Section 1.2.2. Most of the dynamic models consider that consumer share all the information about their demand to the SO. Load scheduling models developed in Chapter 3 also assumed that consumers share complete information to SO. However, consumer privacy is very critical, particularly when all the consumers are connected. In this Chapter, a dynamic load scheduling model is investigated such that consumer privacy and security is not compromised. The objectives of this chapter include:

- Develop a dynamic load scheduling model based on consumer utility function
- Use partial information sharing for load scheduling to respect consumer privacy

The model developed in this chapter is significantly different from the previous models as:

- The consumers send partial information to the SO as against complete information in Chapter 3
- All the consumers are considered from the perspective of SO as against considering just the perspective of industrial consumer in Chapter 4
- Energy forecast for future time slots from renewable sources of energy is considered

5.1 Dynamic Scheduling with different consumer types

In this section, a model for dynamic load scheduling with partial information sharing is developed. SO is the manager and consumers (DLCs) are the participants. In every time slot, the manager observes random events such as real time electricity price released by Macrogrid, electricity harvest from the renewables and the electrical energy stored in the battery. The participants observe random event such as new demands. Information of one player is not known to neither the manager nor other players but the utility function of a consumer is known to SO. The steps in dynamic scheduling with all the consumers are shown in Figure 5.1.

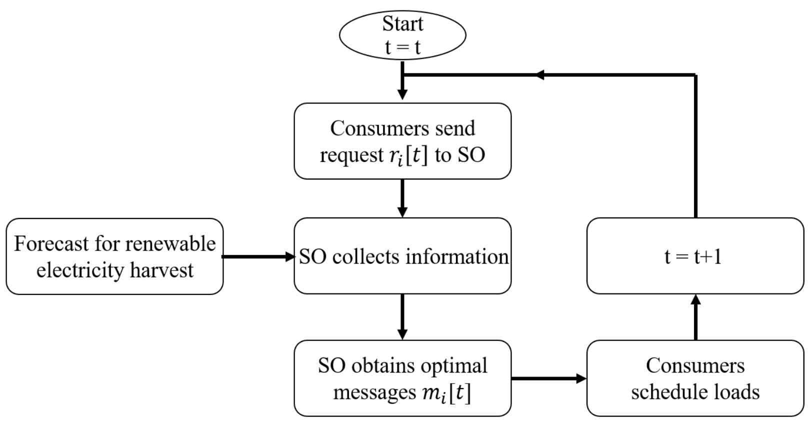


Fig. 5.1. Different steps in dynamic load scheduling with partial information

The consumers share partial information with the manager (SO). In each time slot t, a participant provides the manager with the information of the minimum request and the maximum request. The minimum request in this research is considered to be the non-shift able loads while the maximum request from a consumer is the $X[t]$ obtained in Chapter 4. After receiving information from the consumers, the manager runs an optimization model for finding the optimal supply L^*, sets electricity price p for the participants and the message vector $m[t] = \{m_1[t], m_2[t], ..., m_N[t]\}$ for the consumers. Assuming that the consumers follow the instructions by the manager, $a_i[t] \leq m_i[t], \forall i \in I, \forall t \in T$. After each time slot, based on the consumption by the different consumers, information on virtual queues is updated by the manager. The consumers are charged based on p and total electricity consumed.

5.2 Consumer participation in collaborative Load Scheduling

In this Section, participation of the consumers is characterized for the participating consumers. Time average utility for a consumer i is shown in Equation 5.1. Objective of the dynamic load scheduling problem, shown in Equation 5.2, is to maximize the sum of time average utilities for all the consumers. A participant would participate in load scheduling under certain conditions such that: For a participant i, the time average utility $\bar{u_i}$ is at least as large as the maximum time average utility a participant could achieve if she did not participate in the collaboration, even though she has information on the optimal message from the manager. The condition is shown in Equation 5.3. To ensure that a consumer participates in collaborative load scheduling, time average utility when she plays should be greater than time average utility when she does not participate and therefore has no information about the optimal message from the manager. It is shown in Equation 5.4.

$$\bar{u_i} = \lim_{|T|\to\infty} \frac{1}{|T|} \sum_{t\in T} u_i(a_i[t]) \tag{5.1}$$

$$\text{Maximize} \sum_{i=1}^{N} \bar{u_i} \tag{5.2}$$

$$\lim_{|T|\to\infty} \frac{1}{|T|} \sum_{t\in T} \left(u_i(a_i[t], a_i^{-1}[t]) - u_i(b_i[t], a_i^{-1}[t]) \right) \geq 0 \tag{5.3}$$

$$\lim_{|T|\to\infty} \frac{1}{|T|} \sum_{t\in T} \left(u_i(a_i[t], a_i^{-1}[t]) - u_i(b_i[t]) \right) \geq 0 \tag{5.4}$$

5.3 Dynamic load scheduling

Dynamic load scheduling consists of 4 steps. First, the participants send information about their maximum and minimum demand in each time slot. Second, the manager obtains optimal supply value based on the estimated values of random variables. Third, the manager finds optimal message for each consumer based on Lyapunov optimization. Lastly, the participants schedule the loads based on the message from the manager and update their request for the next time slot.

5.3.1 Request from consumers

In this Chapter, all the three types of consumers with their respective types of loads are considered: Industrial manufacturing facility, residential consumers and commercial buildings. Each consumer sends their minimum and maximum demand to the grid manager.

Electricity requested by each residential building and commercial building and job shop at time t is explained in Chapter 3 in Section 3.2.2 and by job shop is explained in Chapter 4 in Section 4.2. The minimum request from a consumer i is $re_i^{min}[t] = L_i^{ns}[t]$ and maximum request is $re_i^{max}[t] = \sum_{d \in SL_i[t]} L_{i,d}^{sl} + L_i^{ns}[t]$. The minimum and maximum request from a job shop facility is obtained by $re_i^{min}[t] = x^{NS}[t]$ and $re_i^{max}[t] = X[t]$ respectively as explained in Chapter 4. These values are known internally only to the DLCs of the respective consumers but not shared with any other consumer or the SO. In every time slot, the SO sees only $re_i^{min}[t]$ and $re_i^{max}[t]$ from all the consumers, thus only partial information is shared by the consumers to the manager. The consumers DLC may also provide requests at time t in the form of an array containing information about the requirements of future time slots based on their forecast for onsite energy generation.

5.3.2 Optimal supply L^*

It is assumed that SO has an estimation of the future values of electricity harvest from renewables and average demand from all the consumers. This estimation is obtained from the sampling method discussed in Section 4.2.3. Thus, electricity generation from the renewables is assumed to be normally distributed around a time varying mean. This estimation is used to identify how much of total request can be deferred to later time periods. This provides an improvement to the Lyapunov optimization technique. The optimization model for optimal supply is constructed in Equation 5.5 – Equation 5.11. $E_g[t]$, $E_p[t]$ and $E_l[t]$ is the estimated harvest from renewables, estimated electricity price from Macrogrid and and estimated load from the consumers. The estimation is based on sampling using Wesserstein distance as described in Chapter 4. At the end of each day, the problem is solved for optimality and the value of $E_l[t]$ is obtained as a weighted average of the historical optimal values. This also helps in making better load scheduling decisions as the decisions are made based on historically optimal values and it is also able to adapt to new demand pattern. Since a tentative model is considered, storage constraints are neglected in this formulation for future time slots as the objective is to find out how much load can be deferred to future time slots. $\bar{X}[t]$ is the decision variable as how much of the total requested demand should be met at time t where $L^* = \bar{X}[t]$.

$$\text{minimize} \sum_{t'=t}^{t'=t+T^{'}} \left(X[t']E_p[t'] + G[t']E_p[t'] \right) \tag{5.5}$$

$$X\bar{[}t'] + G[t'] = E_l[t'] \quad \forall t' \in \{t+1, ..., t+T^{'}\} \tag{5.6}$$

$$G[t'] \leq E_l[t'] \quad \forall t' \in \{t+1, ..., t+T^{'}\} \tag{5.7}$$

$$\sum_{t'=t}^{t'=t+T'} \left(X\bar{[}t'] + G[t'] \right) = \sum_{t'=t}^{t'=t+T'} E_l[t'] \tag{5.8}$$

$$\bar{X}[t] \geq \sum_{i \in I} re_i^{min}[t'] \quad \forall t \in T \tag{5.9}$$

$$\bar{X}[t] \leq \sum_{i \in I} re_i^{max}[t] \quad \forall t \in T \tag{5.10}$$

$$\bar{X}[t'] \in R^+ \quad \forall t' \in \{t, t+1, ..., t+T^{'}\} \tag{5.11}$$

The objective function to minimize the total cost for period $T^{'}$ is shown in Equation 5.5. The function aims to minimize the cost of an estimated function. The supply and demand balance is shown in Equation 5.6. Equation 5.8 ensures that the consumer requests at time t are scheduled in the following time slots that helps in deciding how much load can be deferred to the future time slots. Equation 5.7

is the resource constraint. The constraint to meet minimum demand is shown in Equation 5.9 while the constraint for maximum demand requested is shown in Equation 5.10. Distribution of the current demand to different time periods is ensured in Equation 5.8. Equation 5.11 is the variable domain constraint.

After finding the optimal supply L^*, electricity price p is proposed by the manager and the consumers will be charged based on p and message $m_i[t]$ (developed in the next section). p is obtained from Equation 5.12.

$$p = \frac{\min(Ha[t], L^*)p_1[t] + \max(0, L^* - H[t])p_2[t]}{L^*} \tag{5.12}$$

5.3.3 Optimal message to the players ($m[t]$)

The utility function of the players is given in Equation 5.13 where c_u is a scaling constant. The first term corresponds to the total electricity cost and the second term corresponds to the priority of the demand fulfillment. The two parts are contradictory in nature. Optimal value of $m_i[t]$ for a consumer is given in Equation 5.14. If a player consumes $b_i[t] > m_i[t]$, she pays for the difference at $p_2[t]$.

$$u_i(a_i[t]) = -\left(pa_i[t] + \frac{c_u re_i^{max}[t]}{2a_i[t]}\right) \tag{5.13}$$

$$a_i^*[t] = \sqrt{\frac{c_u re_i^{max}[t]}{p}} \tag{5.14}$$

The utility of the manager is given in Equation 5.15 that is the total electricity cost.

$$u_o(a[t]) = -\big(X[t]p_1[t] + G[t]p_2[t]\big) \tag{5.15}$$

Their participation constraints are shown in Equation 5.3 and Equation 5.4 respectively are their respective virtual queues Q_i^{CCE} and Q_i^P are shown in Equation 5.16 and Equation 5.17 respectively.

$$Q_i^{CCE}[t+1] = max\left(0, Q_i^{CCE}[t] + p_2[t]\big(m_i[t] - r_i[t]\big) + \phi re_i^{max}[t]\big(\frac{1}{m_i[t]} - \frac{1}{r_i[t]}\big)\right) \tag{5.16}$$

$$Q_i^P[t+1] = max\left(0, Q_i^P[t] + p\big(m_i[t] - r_i[t]\big) + \phi re_i^{max}[t]\big(\frac{1}{m_i[t]} - \frac{1}{r_i[t]}\big)\right) \tag{5.17}$$

The Lyapunov function is shown in Equation 5.18. Using the operators used in Chapter 3 and the drift function, The drift + penalty function for the manager is shown in Equation 5.19. The model is simplified as the terms $\frac{c_u re_i^{max}[t]}{m_i[t]}$ is not included to keep the model simple, linear and convex. Also, since $c_u/re_i^{min}[t]$ does not go to 0 very fast, the term does not vary much and its impact can be adjusted by the parameter value of the parameter V and c_u. In this formulation, the value of c_u may be replaced with $\big(Q_i^{CCE}[t] + Q_i^P[t]\big)$. The penalty term is different from previous models as the penalty is included on the optimal supply L^*. This also removes the problem of scaling and is independent of time.

$$\omega[t] = \frac{1}{2}\Bigg(\sum_{i \in I}\big(Q_i^{CCE}[t]\big)^2\Bigg) + \frac{1}{2}\Bigg(\sum_{i \in I}\big(Q_i^{P}[t]\big)^2\Bigg) \tag{5.18}$$

$$\nabla[t] = C - \Bigg[\sum_{i \in I}\big(Q_i^{CCE}[t]p_2[t]m_i[t] + Q_i^P[t]m_i[t]p\big)\Bigg] + V\big(Z - L^*\big) \tag{5.19}$$

Thus, the manager solves the problem defined in Equation 5.20 $-$ Equation 5.22 as described below. Equation 5.20 is the objective function to be solved at time slot t. Equation 5.21 is the supply demand balance constraint. Equation 5.22 is another supply demand balance constraint that includes the optimal supply L^*. Equation 5.23 is the variable type constraint. Equation 5.24 are the storage constraints defined in Chapter 3.

$$\text{minimize} \quad \nabla[t] \tag{5.20}$$

$$X[t] + G[t] + B[t] = \sum_{i \in I} m_i[t] \tag{5.21}$$

$$Z = X[t] + G[t] + B[t] \tag{5.22}$$

$$Z, m_i[t] \geq 0 \quad \forall t \in T \tag{5.23}$$

$$\text{Equation 3.1} - \text{Equation 3.8} \quad \forall t \in T \tag{5.24}$$

Based on the optimal $m[t]$ and consumption by the consumers, the virtual queues are updated in every time slot. The message vector $m[t]$ is the recommendation to the consumers and they consume according to their actual requirement. It is assumed that the consumers participate in load scheduling and consume power $\leq m_i[t]$.

5.3.4 Scheduling loads by DLCs

The DLCs schedule the loads based on the optimal message from the manager. Since DLCs schedule loads, consumer privacy and security is ensured. The loads are scheduled by each DLC based on the weights of the different loads as developed in Table 3.4 by solving the optimization problem developed in Equation 5.25 − Equation 5.27. The objective function is shown Equation 5.25. The constraint on total consumption, assuming that the participant participates in the collaboration is shown in Equation 5.26. Equation 5.27 is the variable type constraint.

$$\text{maximize} \quad L_{i,d}^{sl} w_d x_d \tag{5.25}$$

$$\sum_{d \in D} L_{i,d}^{sl} x_d \leq m_i[t] \tag{5.26}$$

$$x_d \in \{0, 1\} \quad \forall d \in D \tag{5.27}$$

To compare the performance of the proposed repeated stochastic game, the solution is compared against the benchmark model developed in Section 3.1. Since the model uses the estimation information from future events, this model outperforms the model developed in Section 3.2.

5.4 Simulation results

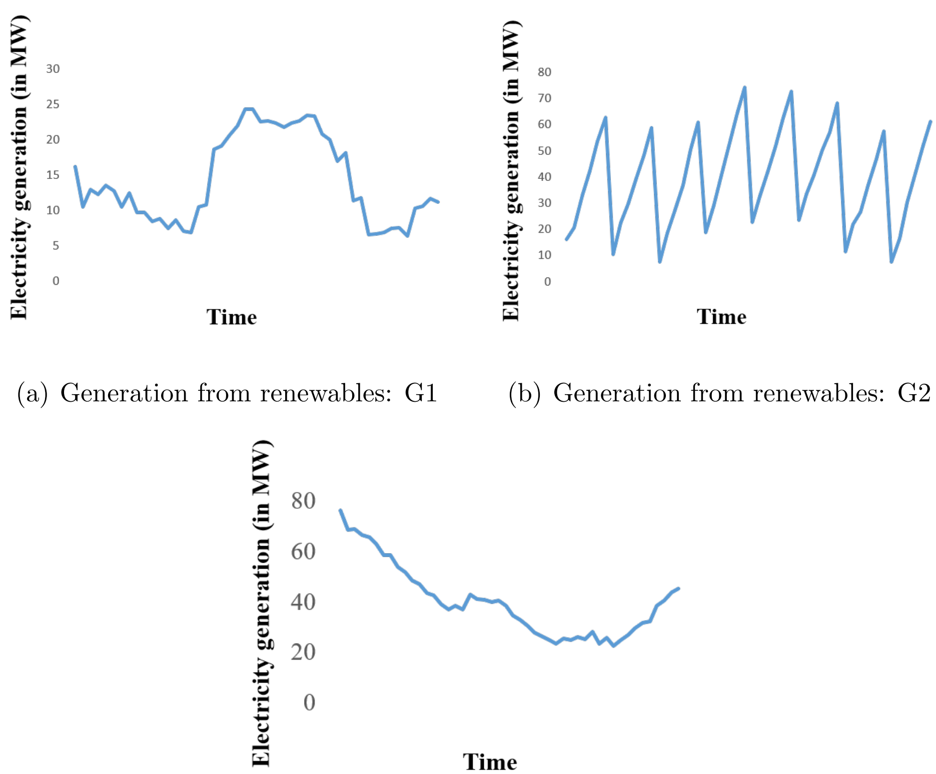
A similar experiment set up is used in this chapter as used in Chapter 3. The performance of the proposed algorithm is compared to the benchmark solutions and the solutions obtained from Lyapunov optimization method in Chapter 3. Since Lyapunov optimization method outperforms different static and individualistic strategy models, results from other models are not shown in this Chapter. The different models are summarized in Table 5.1.

The Lyapunov optimization method in Chapter 3 does not consider the estimation of future values of the random process. The drift + penalty term for Lyapunov method in this Chapter is shown in Equation 5.19.

Table 5.1.
Different load scheduling models in Chapter 4

Model	Definition
Model I	Benchmark scheduling (Section 3.2.1)
Model V	Lyapunov optimization based dynamic scheduling: Chapter 3
Model VI	Proposed Lyapunov optimization using future information

To compare the performance of the these algorithms, three different distributions are used for electricity generation in renewables. The average (expected) value of the three distributions is shown in Figure 5.2. Three different test cases based on the electricity generation from the renewables are denoted as $G1$, $G2$ and, $G3$.



(a) Generation from renewables: G1 (b) Generation from renewables: G2

(c) Generation from renewables: G3

Fig. 5.2. Mean of the electricity generation from renewables

Performance of the different algorithms for different load ratios in terms of the total electricity cost for different test cases are shown in Figure 5.3. The results are obtained for the storage option 1 as mentioned in Table 3.1.

5.4.1 Selecting the value of T'

A sensitivity analysis for T' is performed using Ranking and Selection method in optimization through simulation that takes Bayesian approach. This method is The ranking and selection method selects the best value for T' for the different test cases such that a good average performance is obtained. A sequential sampling methodology is used to find the best value for T'. This is a substitute for the sensitivity analysis performed in Chapter 4. The complete method of is described in Section refappendix1.

The best values of T' for different test cases is shown in Table 5.2. The results are obtained for storage option 1 and load factor 0.25. The optimal values of T' depends on the structure of the problem and distribution of the random variables. The total number of simulation runs (or simulation effort) is kept at 500 over which the allocation of simulation effort is made. In the sequential approach, in each step, only one choice is selected and a random sample is sampled for that choice. Bayesian approach for ranking and selection provides a statistical method to obtain the best choice that is expected to perform good on an average. However, it may not perform well when considering worst case measures.

Table 5.2.
Best values for the parameter T'

Test case	Best value of T'
G1	3,17
G2	14,4
G3	11,4

The intuition behind the algorithm is that if the prior confidence on the prior is low and thus the variance is high, as more and more data is collected, the variance for posterior decreases. In contrast, if the variance for the prior is low, collecting more data does not vary the variance of the posterior distribution.

As the value of T' increases, L^* decreases as it is distributed among the different future time periods $t \leq T'$. Also, as the value of T' decreases, it misses the opportunity of deferring loads to the time of lower cost. Thus, there is trade-off in the selection of T' values. The Bayesian approach of finding the best alternatives among the different values of T' runs the simulation appropriate number of time for different alternatives and finds the alternative with the highest expected value.

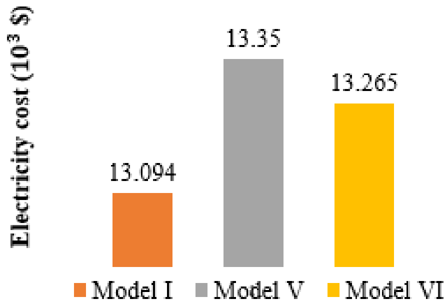
5.4.2 Numerical results

The utility function of the different consumers have been defined using the values as $c_u = 0.8$. The value is obtained such that on an average, the cost term $pa_i[t]$ and the wait term $\frac{c_u re_i^{max}[t]}{2a_i[t]}$ are approximately same for averages as used by Yu and Hong [209]. The results are shown for corresponding optimal values of T' for different test cases as shown in Table 5.2. The results show that the proposed Model VI performs comparable to Model I while outperforming Model V.

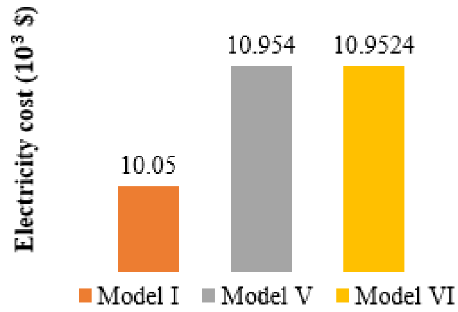
Load profile of the algorithms also shows how the algorithm performs against the benchmark solution. The better the algorithm matches the load profile of the benchmark solution, the better is the performance of the algorithm. Example load profiles for different test cases are shown in Figure 5.4 $-$ Figure 5.6. The results are obtained for storage option 1 and load ratio 0.25.

In test case G1, generation from renewables is low as compared to the demand. In this case, the load profile is determined mainly by the the electricity prices and there is a peak a when the prices is low.

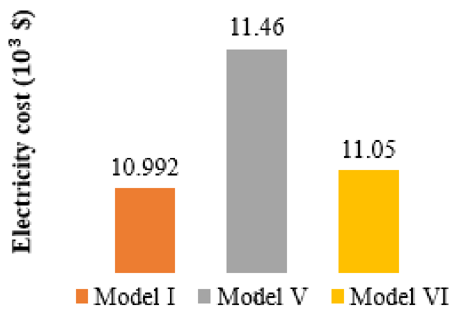
In test case G2, generation from the renewables is periodic with peaks. The proposed Model VI tries to fit the generation profile because generation is more than



(a) Test case G1



(b) Test case G2

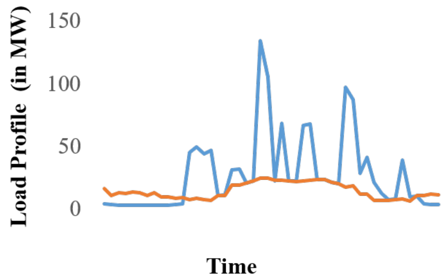


(c) Test case G3

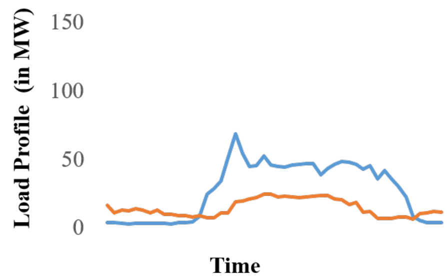
Fig. 5.3. Algorithm performance for different test cases

total demand on an average. In test case G3, the generation is out of sync with the demand and the load profile is determined by the electricity prices as generation is less than demand in most of the time slots.

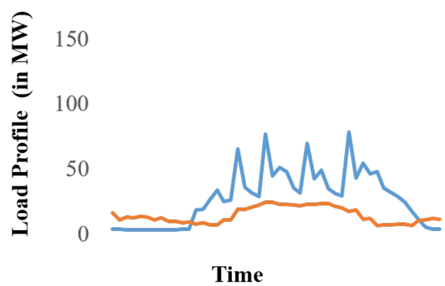
The load profiles for different test cases show that the proposed model performs better in matching the load profile of Model I. Peak load was not considered in above test cases. Peak load is critical as it determines Peak-to-Average Ratio that is critical for grid reliability. One disadvantage with the proposed model is that the loads from different periods may be deferred to a particular time that may lead to a spike in demand at that time slot leading to increased peak-to-average ratio and also increases cost as more electricity is bought from Macrogrid. The peak can be avoided by a peak load constraint in the model. The results with different peak constraints in the different test cases is shown in Figure 5.7. The results are obtained



(a) Load profile for Model I



(b) Load profile for Model V



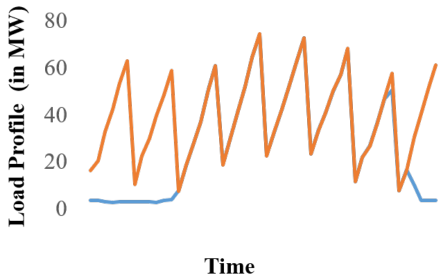
(c) Load profile for Model VI

Fig. 5.4. Example load profile for test case G1

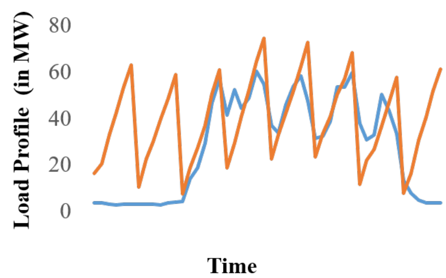
for respective best T' values. The error in Model VI as against Model I with peak load constraint is shown in Table 5.3. The result shows that the solutions from the model are comparable even with partial information. As the peak-to-average ratio is decreased by placing a constraint of 70MW, the total cost for the benchmark solution increases while total cost for Model VI is comparative to that of Model I.

Computation time for Model VI increases as there is an additional step of finding L^*. The comparison for the average computation time is shown in Table 5.4. The results show that the computational time does not increase much for Model VI as against Model V while performing much better than Model I.

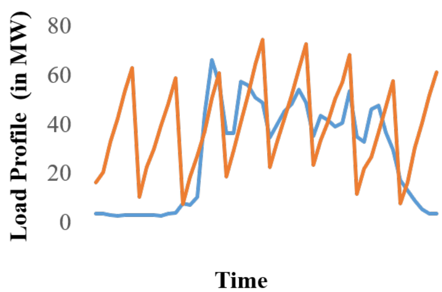
However, the methodology requires a rigorous mathematical study to understand the performance of the algorithm with different stochastic processes with different



(a) Load profile for Model I



(b) Load profile for Model V



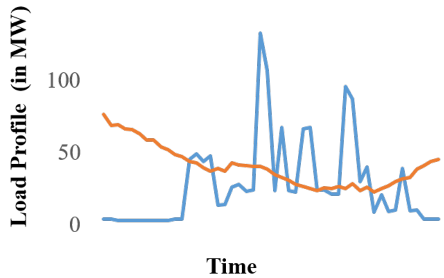
(c) Load profile for Model VI

Fig. 5.5. Example load profile for test case G2

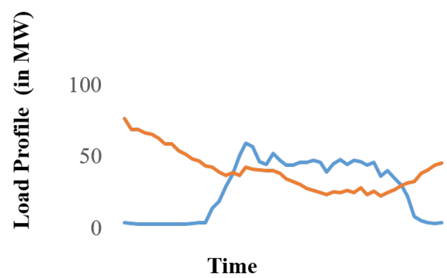
Table 5.3.
Optimality gap in Model VI with peak load constraint

Test case	Error (in %)
G1	1.1
G2	3.3
G3	5.4

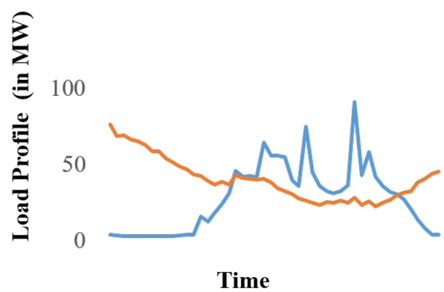
distributions. This work will be carried out in the future to develop a generalized optimization technique for real-time decision making based on Lyapunov Optimization.



(a) Load profile for Model I



(b) Load profile for Model V



(c) Load profile for Model VI

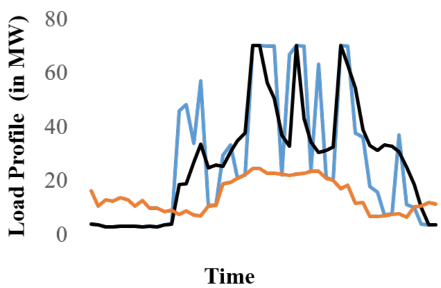
Fig. 5.6. Example load profile for test case G3

Table 5.4.
Computation time for different models

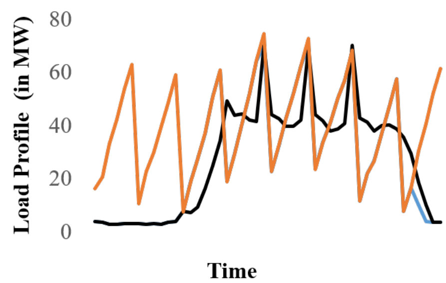
Load ratio	Model I	Model V	Model VI
0.25	110	19	24
0.5	57	18	22
1	31	18	21

5.5 Conclusion and discussion

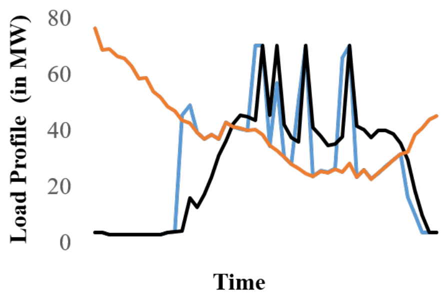
This chapter presents a framework for automated load scheduling based on the concept of collaboration among the consumers. In this model, the consumers reveal



(a) Test case G1



(b) Test case G2



(c) Test case G3

Fig. 5.7. Total electricity cost for Model VI with peak constraints

partial information of their electrical demand to the system operator SO. Based on the request from the consumers, the manager takes control actions using an online algorithm. The model uses information on estimation of future generation from renewables and consumer demand. The notion of consumer utility is established for the load scheduling problem. Different test cases have been simulated to test the performance of the proposed repeated stochastic model. The results show that the proposed method outperforms static approaches, Lyapunov optimization method discussed in Chapter 3 and performs comparatively to the benchmark solutions. The performance of the model improves as against the Lyapunov method when the variability in supply or demand increases.

Since the model is very generalized and it is not restrictive based on any of the structural properties of the load scheduling problem, the model can be extended to different problems such as optimal demand fulfillment in supply chain, spectrum allocation in cloud computing and other problems where the consumer demands can be delayed for optimality. A mathematical approach can be developed to identify the $E_l[t]$ used in this chapter. Stochastic optimization techniques could be used to obtain L^* when the distribution has large variance or the distribution is unknown.

In the next Chapter, the current model is extended is extended to make us of the information given by optimal solutions at the end of the time period based on which the benchmark model is constructed. This information is used to learn a policy that takes dynamic decisions based on the current state, future expectations and what the model has learnt from the environment over time.

6. REINFORCEMENT LEARNING FOR DYNAMIC LOAD SCHEDULING IN A μG

In this Chapter, a reinforcement learning model using guided policy search is presented for dynamic load scheduling in a μG. All the previous models aimed at using a deterministic policy that could achieve asymptotic guarantees for queue stability and bounded cost. The policies require complete state and system dynamics information that is not always available. The performance of the previous models was compared based on the solutions obtained from the benchmark model that has perfect information. This property to obtain the optimal solutions at the end of the period is explored to develop a parametric policy using reinforcement learning based on locally optimal trajectories. The learnt policy does not assume any information on actual state or system dynamics.

The objectives of this chapter is to:

- Propose a framework for reinforcement learning in dynamic load scheduling in smart grid scenario
- Establish and implement a guided policy search for end-to-end real-time decision making in electrical load scheduling

The model developed in this chapter is significantly different from the models developed in previous chapters as:

- An end-to-end parametric policy is learnt using reinforcement learning that makes decisions based on observations rather than the state information
- The problem is converted to a supervised learning problem where data is obtained based on locally optimal trajectories

Note In this Chapter, notations used in the computer science community for reinforcement learning and machine learning are used. Notations for this Chapter are shown in Table 6.1 below.

Table 6.1.
Notations used in reinforcement learning

Symbols	Definition
a_l	output from layer l
b	Batch size
$p(x_{t+1}\|x_t)$	State transition probability
$p(x_1)$	Initial state probability
M	Actions taken based on input X
$l(X, M)$	Loss function for supervised learning
$L1$	L1 norm for regularization
$L2$	L2 norm for regularization
o_t	Observation vector
x_t	State vector
w	Weight vector for the neural network parameters l
X	Input data
Y	Output data
z_l	Input to layer l
τ	Trajectory distribution
μ_l	Batch normalization mean for layer l
σ_l	Batch normalization variance for layer l
$p(o_t\|x_t)$	Observation vector as a distribution of the state vector
$\pi_\theta(\tau)$	Probability distribution of the trajectory when π_θ policy is used
$\pi_\theta(M_t\|x_t)$	System dynamics when action is taken based on the state vector
$\pi_\theta(x_{t+1}\|x_t, M_t)$	System dynamics based on action and the previous state
$\pi_\theta(M_t\|o_t)$	System dynamics when action is taken based on the observation vector

6.1 Demand and capacity sharing among μGs

In this Chapter, a very simple notion of demand and capacity sharing is considered as shown in Figure 6.1. Two μGs are considered that participate in the collaborative demand and capacity sharing. The SO considered in the previous chapters could be the link between a Central Controller (CC) and the participating μGs.

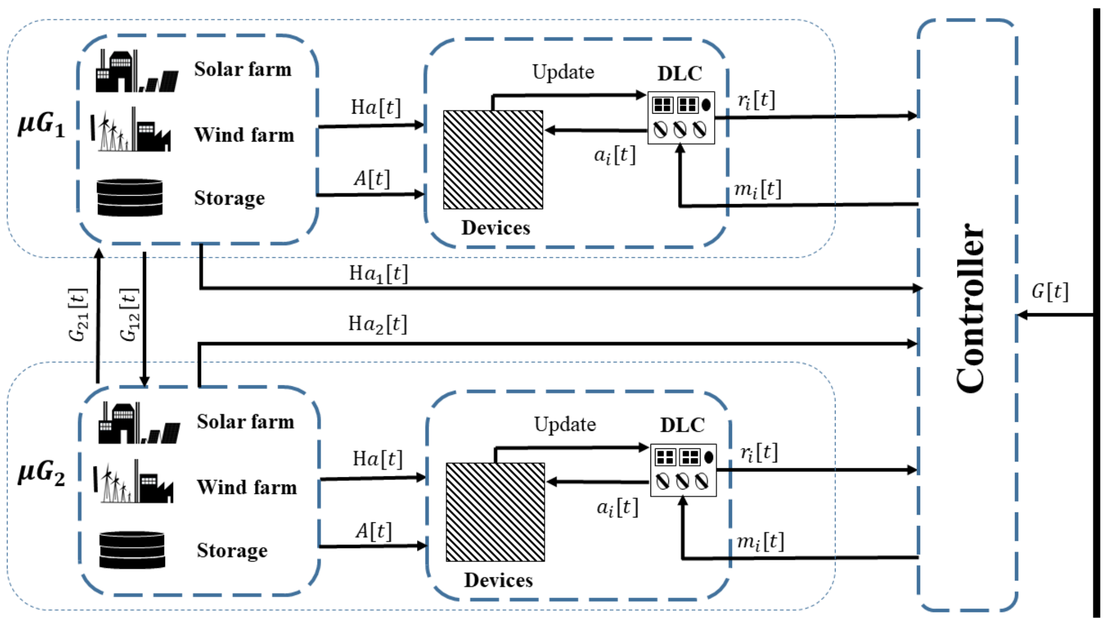


Fig. 6.1. Demand and capacity sharing among μGs

In each time slot t, the consumers send request to the SO. The requests may be send for the immediate time slot as in Chapter 3 or next few time slots as in Chapter 4. In this Chapter, we consider that the consumers send requests for T' time slots based on the demand and on-site energy harvest. The SO sends the demand information from the consumers along with the information of forecasts from renewables in the μG and the energy available in the battery. The central controller then sends messages to the consumers on how much electricity can be consumed in time slot t. If the consumer uses more electricity, they have to pay for the extra electricity consumed at higher price.

The SO sends messages received from the central controller to the DLC's that schedules the different electrical loads based on their weights of the different loads. The weights are given as shown in Table 3.4. In each time slot t, an array $SL_i[t]$ is maintained that contains information on the per time unit power demand and weight of that load at time slot t as shown in Equation 3.46. $SL_i[t]$ contains information of the loads that have been requested but not yet scheduled. Then the loads are scheduled such that the power constraints are not violated for the t^{th} and the next T' time slots. The heuristic to generate electricity demand request to SO by DLCs' is given by Heuristic 6.1 and the heuristic to schedule the loads after getting message from the SO is given by Heuristic 6.2.

Heuristic 6.1: Generating electricity request by DLCs

1. Define $re_i^{min}[t] = 0$ and $re_i^{max}[t] = 0 \quad \forall t \in \{t, t+1, ..., t+T'\}$
2. Define $SL_i[t]$ as explained in Equation 3.46
3. $re_i^{min}[t] = x_i^{ns}[t]$ and $re_i^{min}[t'] = E\big[x_i^{ns}[t]\big] \forall t \in \{t+1, t+2, ..., t+T'\}$
4. $re_i^{max}[t] = re_i^{min}[t]$
5. Update $re_i^{max}[t, t+1, ..., t+T']$ by adding the power demand of all the different loads for the time slot t for which the load is to be scheduled. Schedule the loads such that the power is balanced in the future T' time slots.

Based on the total electricity requested by the consumers and the realization of different stochastic variables such as electricity harvest from the renewables, final consumption by the consumers and the electricity prices, the optimal messages for each consumer and for each time slot can be obtained by solving the optimization problem in Equation 6.1 - Equation 6.15 at the end of the planning period. Since we consider an infinite horizon problem, the optimization problem is solved at end of every week. This solution is locally optimal for that week only and it is considered as the locally optimal trajectory for reinforcement learning as discussed in the next two

Heuristic 6.2: Electricity load scheduling by DLCs

1. Read messages $m_i[t, t+1, ..., t+T']$ from SO
2. Sort loads in $SL_i[t]$ according to their weights
3. Start scheduling the load based on their weights. Schedule the load in the time slots (starting from t) such that the total power consumption does not exceed the $m_i[t']$ for every time slots in which that load is run. This is based on the assumption that the consumers participate and do not consume more than $m_i[t]$
4. Add unscheduled loads to the array $SL_i[t+1]$

sections. Since a stationary distribution is assumed, two weeks can be considered as independent from each other.

Equation 6.1 - Equation 6.15 solve the optimization problem for the week $\bar{w}$ from time slots $\bar{N}$ to $\bar{N} + t \,\text{x}\, 2 \,\text{x}\, 24 \,\text{x}\, 7$ where $\bar{N}$ is a multiple of $2 \,\text{x}\, 24 \,\text{x}\, 7$ and the time slots make the set $\bar{T}$. 2 represents the time slots of 30 minutes, 24 represents the 24 hours in a day and 7 represents the 7 days in a week. In this Chapter, peak demand charge is not considered. However, it can be included in the model easily by adding a peak constraint as shown in Chapter 3. Three different types of energy sources are considered and two μGs are considered as shown in Figure 6.1. Let $\bar{U}$ represents the set of μGs. The notations are similar as in Chapter 3 but for every variable, a bar is added to differentiate the model in this Chapter. For example, $X[t]$ is replaced by $\bar{X}_u[t]$ for μG_u. Energy sources include wind farms in μG_u and solar farms in μG_v. If the demand cannot be met by the battery or energy storage, the μGs can purchase electricity from central grid at prices $p_2[t]$. The cost of electricity is considered same as in Chapter 3. μG_u can give $\bar{X}_{u,v}[t]$ to μG_v at the rate of $p_5[t] \geq p_1[t]$.

The objective function of minimizing cost is given by Equation 6.1. To maintain fairness amongst the participating consumers, Equation 6.2-Equation 6.3 is developed that ensures that the delay is proportional to the total consumption by a consumer. Equation 6.2 defines f_i as the total delay in fulfillment of the demands over the time slots $\bar{T}$. It is assumed that all the demands have to be met as the loads have a deadline by which it has to be completed. For any load, let t_d be the time at which that particular load was scheduled/completed. Based on the different types of loads, subscripts are added in t_d as shown in Equation 6.2. Equation 6.3 ensures that the fairness is maintained between two different consumers in a μGs. This fairness is maintained across all the consumers in a μG such that the advantage/ disadvantage of having more/fewer renewable resources are directly given to the consumers of that μG and not spread across μGs. The notations are similar to that discussed in Chapter 3 but subscripts are added to keep the notation succinct. For example, A_{max} in Chapter 3 is replaced by $A_{max,u}$ for the battery capacity for μG_u. The demand and supply matching is shown in Equation 6.4-Equation 6.6. Equation 6.4 is the total consumption in a μG in time slot t. Equation 6.5 ensures that the the non-shift able loads of consumers are met in every time slot t. Equation 6.6 ensures that the total consumption by the consumers in μG equals to the total supply in that time horizon. The optimization problem is solved from a central controllers perspective, hence only the information on consumption is known but no information is available about the types of loads. The battery constraints for μG_u is shown in Equation 6.7-Equation 6.14. These constraints are similar to that explained in Equation 3.1-Equation 3.8. Since the problem is locally optimal, a threshold $T\bar{H}$ is placed such that the energy level of the battery does not go beyond this level as shown in Equation 6.13. This constraint also ensures safety of the battery. The domain constraints are shown in Equation 6.15.

$$\text{Minimize} \qquad \sum_{t\in\bar{T}}\left(\sum_{u\in U}\left(p_1[t]\bar{X}_u[t]+p_2[t]\bar{G}_u[t]\right)+\sum_{u,v\in U}p_3[t]\bar{X}_{u,v}[t]\right) \tag{6.1}$$

$$f_{i,u}=\sum_{r\in R_i}\left(t_{d,r,i}-t_{i,r}^{ts,min}\right)+\sum_{r\in R_i}\left(t_{d,h,i}-t_{i,r}^{ps,min}\right) \qquad \forall i\in I_u,\forall u\in U \tag{6.2}$$

$$\frac{f_{i,u}}{f_{i',u}}=\frac{\sum_{t\in\bar{T}}a_{i,u}[t]}{\sum_{t\in\bar{T}}a_{i',u}[t]} \qquad \forall i,i'\in\mu G_u,\forall u\in U \tag{6.3}$$

$$\bar{G}_u[t]+\bar{X}_u[t]+\sum_{v\in U,v\neq u}\bar{X}_{v,u}[t]=\sum_{i\in\bar{I}_u}a_{i,u}[t]+Y_u[t]-B_u[t] \quad \forall u\in\bar{U},\forall t\in\bar{T} \tag{6.4}$$

$$\bar{G}_u[t] + \bar{X}_u[t] + \sum_{v \in U, v \neq u} \bar{X}_{v,u}[t] \geq \sum_{\bar{I}_u} x_i^{NS}[t] \quad \forall u \in U, \forall t \in \bar{T} \tag{6.5}$$

$$\sum_{t \in \bar{T}} \left(\bar{G}_u[t] + \bar{X}_u[t] + \sum_{v \in U, v \neq u} \bar{X}_{v,u}[t] \right) = \sum_{t \in \bar{T}} \left(\sum_{i \in \bar{I}_u} a_{i,u}[t] \right) \quad \forall u \in U \tag{6.6}$$

$$\bar{A}_u[t+1] = \eta \bar{A}_u[t] + \bar{Y}_u[t] - \bar{B}_u[t] \qquad \forall t \in \{1, 2, ..., T\}, \forall u \in \bar{U} \tag{6.7}$$

$$\bar{B}_u[t] \leq \bar{A}_u[t] \qquad \forall t \in \{1, 2, ..., T\}, \forall u \in \bar{U} \tag{6.8}$$

$$\bar{B}_u[t] \leq \bar{B}_{max,u} \qquad \forall t \in \{1, 2, ..., T\}, \forall u \in \bar{U} \tag{6.9}$$

$$\bar{Y}_u[t] \leq \bar{Y}_{max,u} \qquad \forall t \in \{1, 2, ..., T\}, \forall u \in \bar{U} \tag{6.10}$$

$$\bar{A}_u[t] \leq \bar{A}_{max,u} \qquad \forall t \in \{1, 2, ..., T\}, \forall u \in \bar{U} \tag{6.11}$$

$$\bar{B}_{max,u}, \bar{Y}_{max,u} \leq \bar{A}_{max,u} \qquad \forall t \in \{1, 2, ..., T\}, \forall u \in \bar{U} \tag{6.12}$$

$$\bar{A}_u[|\bar{T}|] \geq T\bar{H}_u \qquad \forall u \in \bar{U} \tag{6.13}$$

$$\bar{A}_u[t], \bar{B}_u[t], \bar{Y}_u[t] \in R^+ \qquad \forall t \in \{1, 2, ..., T\}, \forall u \in \bar{U} \tag{6.14}$$

$$\bar{G}_u[t], \bar{X}_u[t], \bar{X}_{u,v}[t] \in R^+ \quad \forall t \in \bar{T}, \forall u, v \in \bar{U} \tag{6.15}$$

6.2 Guided Policy Search

In this Chapter, a deterministic policy for load scheduling is developed by converting the problem of policy search into a supervised learning problem by iteratively constructing the training data based on the optimal solutions obtained at the week as discussed in the previous Section. This method is known as the guided policy search as the policy is guided by the optimal trajectory (optimal messages in this Chapter) based on the observations (different indicators of state) rather than the state as full state information is not observable in real time.

It is based on locally optimal trajectories as the training data is generated by solving the problem for one week and not the entire planning horizon. This this method is known as guided policy search with locally optimally trajectories. The training set for supervised learning can be constructed using known system dynamics as in the previous Section.

In this Chapter, an episode is defined as the time horizon for a week for which the optimal values can be obtained at the end of the week. The goal of guided policy search is to find a policy $\pi_\theta(M_t|o_t)$ where M_t is the vector of optimal message vector and o_t is the observation vector. The system dynamics given by $p(x_{t+1}|x_t)$ is not known at time t but the decisions are made based on o_t where o_t is stochastic consequence of x_t. In general, $\pi_\theta(M_t|o_t)$ is the distribution of actions under the policy conditioned on state at time t. The probability over the trajectory τ is given by Equation 6.16. However, since the policy is conditioned on observation rather than the state, the obtained policy distribution is given by Equation 6.17

$$\pi_\theta(\tau) = p(x_1) \prod_{t=1}^{\tau} \pi_\theta(M_t|x_t) p(x_{t+1}|x_t, M_t) \tag{6.16}$$

$$\pi_\theta(M_t|x_t) = \int \pi_\theta(M_t|o_t) p(o_t|x_t) do_t \tag{6.17}$$

If a loss function in supervised learning is given by $l(x_t, M_T)$, objective of the guided policy search method is to minimize the total expected loss as given in Equation 6.18.

$$\text{Min} \quad E_{\pi_\theta}\big[\sum_{t=1}^{\tau} l(x_t, M_t)\big] \tag{6.18}$$

Since the state variables are not known at time t, the loss function cannot be obtained. However, using guided policy search, the loss function can be estimated using the loss function shown in Equation 6.20.

This method consists of two steps. In the first step, training dataset is obtained by solving the problem optimally at the end of the week when all the stochastic variables for each of the time slots t in that week are realized. In each time slot t, the observation vector is saved. Thus at the end of the week, the observation vector and the optimal values are known for each of the time slots. In the second step, neural networks is used to learn the policy (or a mapping function) that maps the observations from each time slot t to the optimal messages obtained for that time slot. These two steps are not sufficient in obtaining good results as explained in the next section.

6.3 Locally optimal trajectory

Converting the policy search problem into a supervised learning problem memorizes the the data but does not learn the policy mentioned in Heuristic 2 as this policy is not visible to the reinforcement learning. This fits well into the model so as to maintain the consumer privacy so that the loads are scheduled with partial information sharing. To learn the policy, an iterative approach is considered as shown in Figure 6.2.

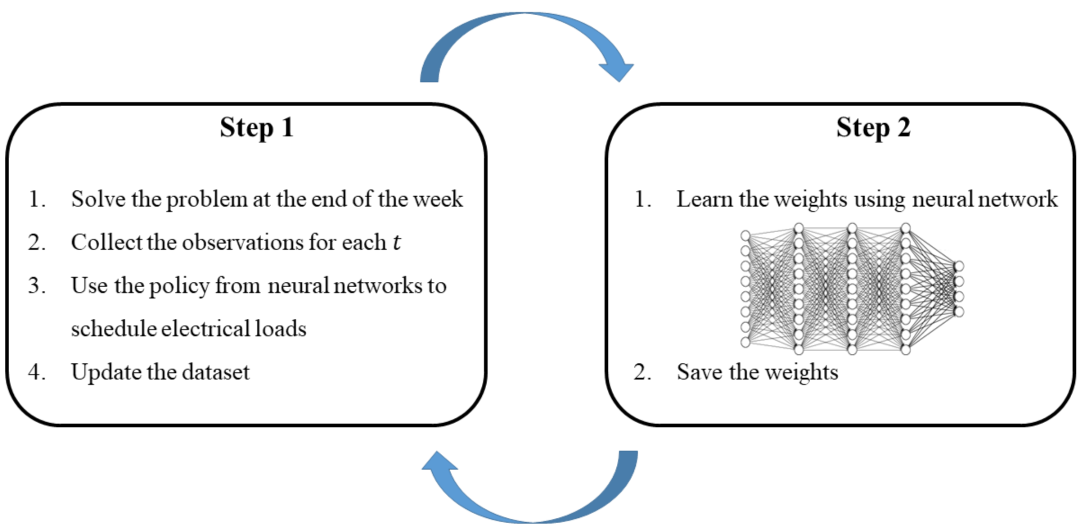


Fig. 6.2. Iterative procedure for the policy search

When starting the iterative procedure, the messages $m_i[t]$ from the central controller to consumer i is given by $re_i^{min}[t] + uniform(0.25, 1)\big(re_i^{max}[t] - re_i^{min}[t]\big)$. This messages are used to schedule loads and hence generate the data for the first iteration for training the neural network. This iterative procedure ensures that the method not only performs well in learning from data, but also learns the policy to take action when unforeseen data is observed. The performance increases substantially on using the iterative step as against increasing the number of data points collection.

The policy defined by $\pi_\theta(M_t|o_t)$ provides the guidelines to the central controller in real time. System dynamics distribution given by $p(M_t|x_t)$ governs how the system evolves when the true system information is known. By using the iterative procedure, the behavior of $\pi_\theta(M_t|o_t)$ and $p(M_t|x_t)$ converges. This ensures that the policy understands the system dynamics as well as the optimal messages that could be been provided to consumers in each time slot based on the observation information that could potentially represent the system state information.

This method does not learn the policy parameters directly as done in Q-learning or traditional reinforcement learning. These methods require that all the states and actions should be seen by the model during the training period. This is practically impossible when the state space could be continuous or the problem size in large. The proposed method scales gracefully with the increase in size. It uses locally trajectories as guideline to learn "what we could have done better" decisions. Using these guidelines also enables that the learning part could be done with much less data as the target value is known and during each iteration step the parameter values (weights in neural networks) are saved.

6.4 Neural network for Policy Approximation

A neural network (multi-layer perceptron) is used to learn the deterministic policy . The architecture example used in this research is shown in Figure 6.3. $h1, h2$ and $h3$ are the number of hidden neurons (nodes) in each layer. In this research, 8 hidden layers have been considered to learn the mapping function. The rectified linear unit (ReLU) activation shown in Equation 6.19 function is used for mapping inputs to the output. z is the output from a layer, w is the weight and a is the linear combination of outputs from previous hidden units and the weights. The input vector (X) to the neural network includes:

- maximum electricity demand $re_i^{max}[t]$ from each consumer in a μG for the next T' time slots.
- forecast of electricity harvest from the renewables at time t

The output vector Y contains:

- the optimal messages information $m_i[t]$ for the next T' time slots for each of the consumer
- how much renewable energy should be used by the μG by itself and how much should it be given to the other collaborating μGs. In this Chapter, since 2 μGs are considered, it creates a vector of size 4
- storage at the end of the time slot

The Y vector is obtained after solving the optimization problem while the X vector is obtained by storing the values of data points during the simulation run. The output can be used to back calculate $\bar{X}_u[t]$ and $\bar{G}_u[t]$ from the output. In case of infeasibility, energy usage for a μG is given priority before allowing the other μG to draw electricity from it.

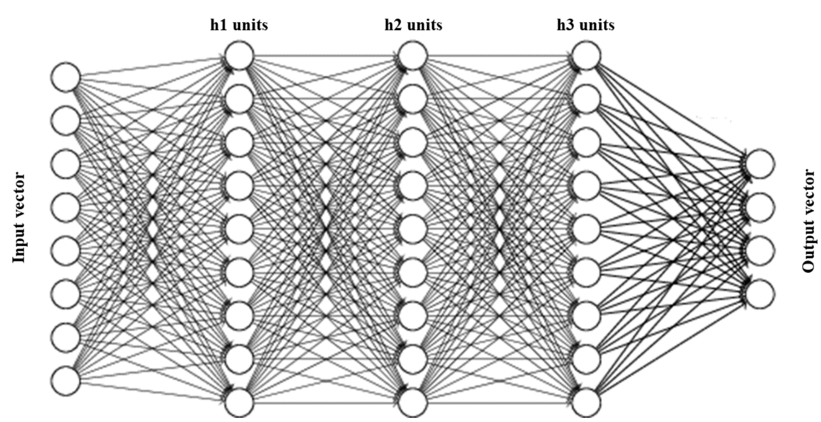


Fig. 6.3. Fully connected neural network architecture for optimal policy learning

$$z_l = \max(0, a) \quad \text{where} \quad a = w^T z_{l-1} \tag{6.19}$$

Along with the input vector, the vector of $re_i^{min}[t]$ is maintained to such that the non-deferrable electricity loads for time t can be scheduled at time t. This information is the part of the behaviour policy that is different from the target policy.

Neural network models have different hyper-parameters and the values of these hyper - parameters used in this research are shown in Table 6.2. The value of the hyper - parameters are obtained based on Bayesian Optimization for Hyper parameters as discussed in [265]. To obtain the best hyper parameter values, bayesopt library for python is used ([266]). The hyper parameters are selected by running the experiments for 100000 iterations and the model with lowest error is selected. Neural networks is not the focus of this research so the details of these hyper-parameters are omitted here.

Table 6.2.
Hyper - parameters for neural network

Hyper - parameter	Value
Depth (number of layers)	3
Number of iterations	500,000
Activation function	ReLU
Regularization	L1 norm
Weight of Regularization	0.01
Optimization	ADAM optimizer
Learning rate	0.05
Batch size	32
Batch normalization	True
epsilon (Batch normalization)	0.1
Training set proportion	0.8

The loss function for the neural network is shown in Equation 6.20. y and z_8 are in vector form. y is the response while z_8 is the output from the final layer. Squared loss function is used as it is a convex function and batch normalization further smooths out the function for optimization. Squared loss is considered as if the message vector (output from the machine learning) is higher than the optimal, the the electricity cost increases. If it is lower, the demand from the consumer cannot be met and hence the service quality decreases. L1 regularization and ReLU is used to make the network sparse so that it learns faster.

$$loss = (y - z_3)^2 + 0.01 \,\mathrm{x} \sum_{l=1}^{3} |W_i| \tag{6.20}$$

Batch normalization is used to increase the speed of learning as it deals with covariate shift in the inputs and outputs of the different layers and makes the optimization function smooth. The batch normalization discussed by Ioffe and Szegedy ([267]) is applied in the second and third layer of hidden units since the data is not i.i.d as assumed in traditional machine learning. Since the data in a batch may be sampled from different hour of the day, the variance is large and intuitively (as well as based on the performance of the model), the performance worsens if the batch is normalized in the first layer. However, the performance improves if the batch is normalized in the rest of the layers. Also, the ReLU activation is applied both before and after the activation function. In the original application of batch normalization, ReLU is applied after batch normalization, but the results in this Chapter show that the model performs better if the ReLU function is applied before batch normalization. Also, the performance improves if ReLU activation is applied both before and after batch normalization. This result could also be a potential area of research for understanding the performance of the batch normalization method. In the next Section, we brief introduction of existing reinforcement learning techniques is provided to show the relationship with the current methodology proposed in this Chapter.

6.5 Why neural network for policy?

Deep neural networks could theoretically may any input to any output given they are drawn from the same distribution. In this research, we assume that data from weeks are i.i.d. and thus we can assume that the optimal values of the output are drawn from the same distribution. Neural networks, however, lack interpretation and is mostly a black box. To handle this, different interpret able machine learning algorithms have been tested. These algorithms perform very poorly as against neural network policy. Thus, in this research, neural networks were used. Some of the algorithms tested in this research and their potential reasons for failures include:

- linear regression models are well known for their simplicity and interpretability. However, the mapping of state variables to the optimal values is highly non-linear. Using kernel tricks and feature engineering is very difficult as the functional form that is linear to output variables is not known.

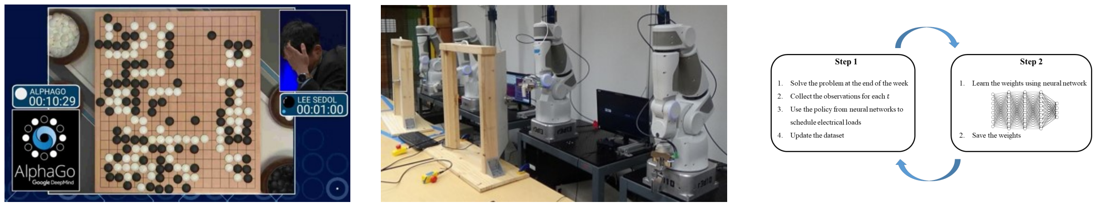
- K-nearest neighbour and ensemble methods like generalized boosting model (eXtreme Gradient Boosted Trees, a.k.a, XGBoost) divide the space and take the average as the output. These methods can map the non-linearity but these are based on averaging and cannot extrapolate outside the training set. In dynamic load scheduling, the averaging also inhibits and limits the importance of time based variables. Since the state variables are presented in the form of time based variables, averaging does not perform well.

A research direction is to develop different models for learning the mapping function from the input to the output. Different machine learning or reinforcement learning algorithms could be developed instead of using neural networks. Deep neural networks are the current state of the art that provide end-to-end mapping capability without much feature engineering (e.g. w developed time based normalization for data normalization). As the systems become complex, feature engineering would become too complex and we need end-to-end methods. Thus, in this research, neural networks are used to learn the scheduling policy.

6.6 Relationship with different Reinforcement Learning Models

The proposed iterative procedure draws heavy motivation from different reinforcement learning methods discussed in literature. off policy, guided policy search, deterministic policy gradient, The iterative procedure in Algorithm 1 is inspired from the Guided Policy Search discussed by Levine et. al ([246] and [225]). Optimal solutions for $m_i[t]$ obtained by solving the problem at the end of the week can be considered as the locally optimal trajectories. The proposed iterative algorithm is model free as it does not consider any information or knowledge on state transitions. A function approximation is used that learns policies in high dimension continuous action and state spaces. The proposed method is off-target policy as the target policy is different from the behavior policy that is used to generate the data for the supervised learning [268]. The policy for load scheduling developed in this work is a deterministic policy and a deterministic policy gradient is used [247] as the target policy is learnt by using an off policy critic. In the supervised learning, the response variables provide stable targets that are not shifting with time, which makes the function approximation stable and robust. The algorithm can also be considered as an actor critic algorithm where the optimal values of $m_i[t]$ act as a critic to the approximating function of the actor. Moreover, the algorithm does not need to learn the critic function, rather it is obtained from the convex optimization. This significantly improves the speed of the reinforcement learning algorithm. Batch normalization is used in all layers except the first layers due to time dependency of the predictor variables. Moreover, for stable learning, batch learning is used and the samples in the batch can be assumed to be independent as they have been normalized as discussed in Algorithm 3.

We discuss an analogy with the current state of reinforcement learning with two examples and explain why the proposed methodology is different from the existing reinforcement learning models. While reinforcement learning is mostly used in computer games with graphics as inputs, their state representation is simple and straight forward. In the following figure, three different test cases are shown. We compare and contrast them with respect to the dynamic load scheduling model developed in this Chapter.



(a) DeepMind AlphaGo using deep neural network (b) Robotic control for lifting and placing objects (c) Dynamic load scheduling in μG

Fig. 6.4. Examples of application of reinforcement learning for building analogy in Table 6.3

Direct application of deep learning techniques achieves poor result and requires model specific adjustment. First, the predictor variables are normalized using mean and variance for its corresponding time of the day as described in Algorithm 2. Second, a sampling based forecasting method using Wasserstein Distance is used to predict energy from renewables as described in Algorithm 3. Batch normalization is also used in the hidden layers.

The architecture of the neural network is kept the same for the iterative step shown in Figure 6.2. The iterative step helps the neural network to understand the policy used for load scheduling shown in Equation 6.1 - Equation 6.15. At each step in the iteration process, the weights are saved and the supervised learning starts from the saved weights. This makes the learning process faster and the weights are not learnt from the scratch. This helps in speeding up the process of learning the weights and the policy, thus reducing the time to test and validate the model.

6.7 Simulation Results

As discussed in Section 6.1, two μGs are considered and each μG has a commercial building as discussed in Chapter 3. Only non-production line loads are considered

Table 6.3.
Comparing existing applications of reinforcement learning

	AlphaGo	Robotic Control	Load Scheduling
State Representation	Image of the Board at t^{th} turn	Image (video) of robot camera at t^{th} action time instance	Demand request and forecast from renewables at time t
State size	Discrete (high cardinality)	Continuous set (infinite)	Continuous set (infinite)
Reward Calculation	Reward based on win/lose at the end of the game	Does not calculate reward	Does not calculate reward
Method Used	Monte Carlo Tree Search and neural network	Neural network based policy search	Neural network based scheduling actions
Use optimal solutions	No, but consider value functions	Yes (locally optimal)	Yes (locally optimal - one week's decisions)
Value function calculation	Yes	No	No
Using forecast to represent state	No	No	Yes
Mathematical optimization for response variables	No	No	Yes
Iterative procedure for internal policy	No	Yes	Yes

Heuristics 6.3: Normalization of predictor variables

1. During warm up period:
   1.a For each t, initialize $D[t] \leftarrow \phi$
   1.b $D[t] \leftarrow D[t] + \{o[t]\}$
2. For each t:
   2.a Use K means to divide $D[t]$ into M_t clusters using Mahalanobis distance
   2.b Store the mean and the covariance structure of each cluster
3. While obtaining $m_i[t]$:
   3.a For each $o[t]$, find the cluster which it belongs to
   3.b Normalize $o[t]$ according to that cluster

Heuristics 6.4: Sampling for forecasting energy from Wind mill

1. Similar to Algorithm 2, clusters are maintained for energy from wind mill for each t
2. For each time t, collect power from wind mill for previous F time slots in array T_F
3. Find the corresponding cluster. Select predictions S_t in that cluster randomly.
4. Find the weighted mean based on the distance of S_t from the T_F

so that the policy can be developed based on exact solutions as the problem can be solved to optimality when considering buildings loads.

Error is defined as the difference from the optimal solution. The results are shown for the electricity cost of 1 week. Figure 6.5 and Figure 6.6 shows the decrease in error with the number of iterations. The model is run 3 different times from scratch for 200 iterations. Two different cases are shown. In Case I, the supervised learning performed better than Case II as in Case I, the size of the μGs were kept similar. In Case II, the sizes were different. Also, in both the cases, two more models are considered where the μGs may or may not collaborate in sharing demand and energy resources. Because of the similar size of the μGs in case I, the input and output variables of the supervised learning consists of a smoother function. Thus, the method could perform better if the load is balanced between collaborating μGs.

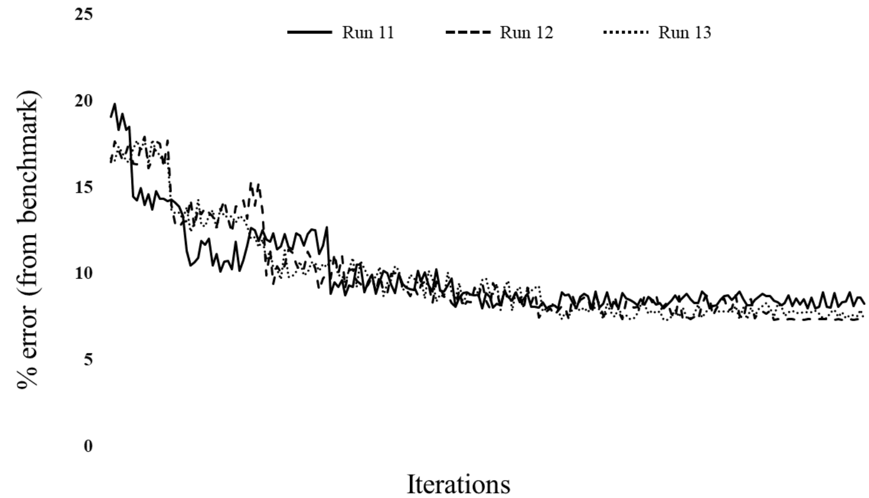
Three different runs (sample path of errors) have been shown for all the four cases. The iteration is restricted to 200. This number is selected as the three different runs seems to converge to a similar error rate. This ensures that the performance of the model is not dependent on the sample path, thus it ensures the stability of the model. The error rate decreases with the number of iterations as the model could learn the values of θ and understand the policy. It does not always decrease as the data is sampled randomly and also weight updates may lead to worse solutions. This helps in making the performance of the algorithm more robust to changes and multiple start times at the start of the week. The error reduction with iteration is shown for the case when the participating μG collaborate and when the μGs do not collaborate with each other in Figure 6.5 and Figure 6.6 respectively.

Example of mapping function value from two time slots selected at random is shown in Figure 6.7. It shows that the supervised learning is able to map the input to the output accurately. Also, the performance for Case I is better as against that for Case II when the μG sizes are similar because of a more smoother function. During the experiments, whenever a new case study is developed, previously saved values of θ are used to generate the new case study. The performance shows that the weights are very robust to the inputs.

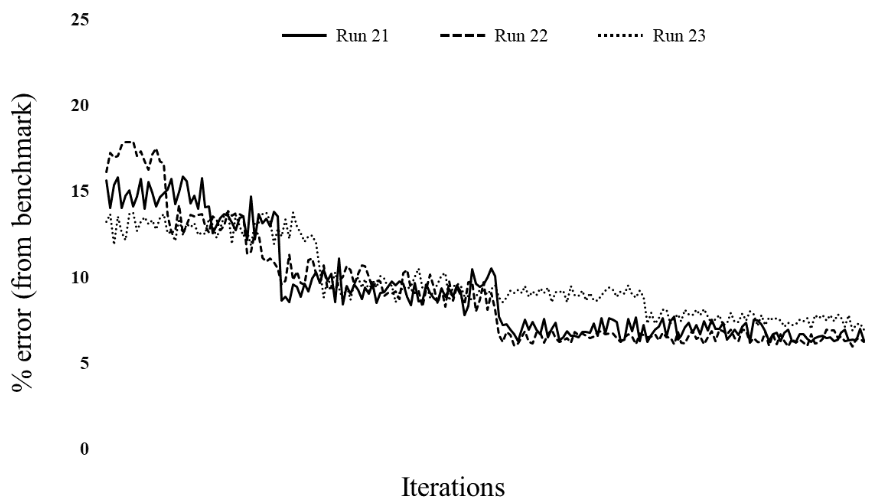
Table 6.4 shows the results for collaboration between 2 μGs. In the majority of the cases, the collaboration between the μGs help them reduce their operating cost as against no collaboration. The results are obtained as average of 30 simulation runs of 1 week. When the μG sizes are similar, the algorithm performs better for no collaboration as compared to collaboration in Case II with μGs of different sizes.

Table 6.4.
Results in terms of percentage error from benchmark

Case	Error: Collaboration	Error: No Collaboration
Case I	6.59	7.22
Case II	7.95	8.08



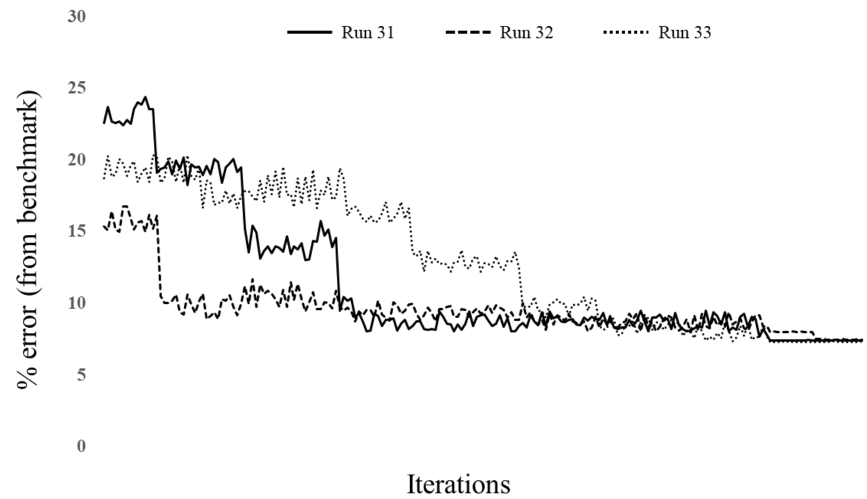
(a) Error with iterations: Case I



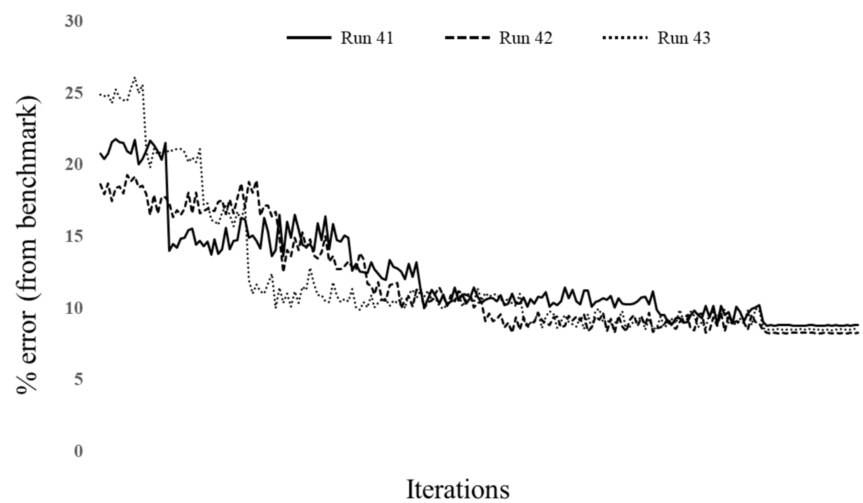
(b) Error with iterations: Case II

Fig. 6.5. Error with iterations for Collaborative μGs

The results show that the no collaboration in Case I performs better than collaboration and no collaboration in Case II. Intuitively, collaboration should outperform no collaboration. These results show that based on the structure of the problem under consideration, the performance might differ. Thus, the objective and nature of



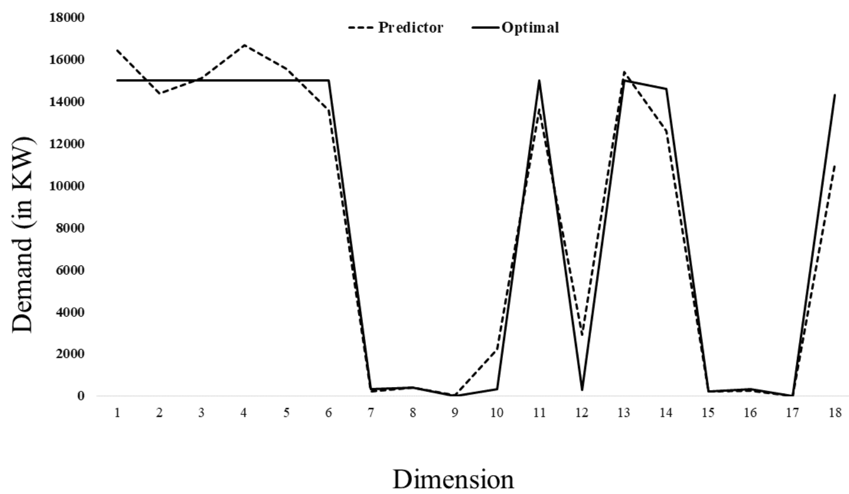
(a) Error with iterations: Case I



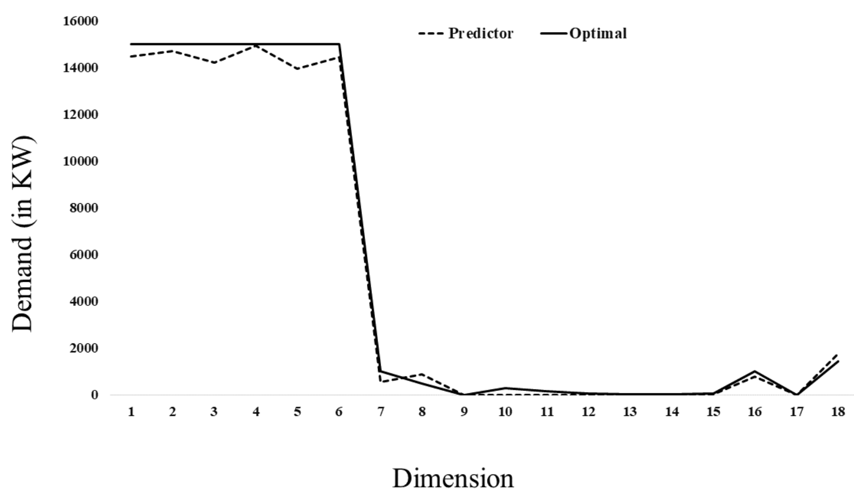
(b) Error with iterations: Case II

Fig. 6.6. Error with iterations for Non - collaborative μGs

the μG should be known in advance for better performance of the proposed reinforcement learning model. The results show promising potential for using reinforcement learning in dynamic load scheduling. Most importantly, this is an end-to-end method that does not require model specific fine tuning of the hyperparameters. This en-



(a) Case I: Predicted vs optimal message



(b) Case II: Predicted vs optimal message

Fig. 6.7. Response and Optimal message $m_i[t]$ for two different cases

ables usage of the existing methodology directly into other problem structures. This Chapter further provides a framework on how stochastic optimization problem can be integrated with machine learning (deep learning) for optimization problems. The results can be improved by increasing the number of layers and optimizing the machine learning method used in this Chapter.

6.8 Conclusions and discussions

This research develops a new framework of integrating operations research with reinforcement learning through optimal guided policy search. In this Chapter, local guided policies are obtained by solving the load scheduling problem at the end of the week, thus by realizing all the stochastic variables involved in the model. At every time slot t, indicators that can help in making decisions are stored e.g. forecast from renewable, unscheduled/partially completed electrical loads, and, information on non-shiftable loads. The guided policy search based reinforcement learning converts the sequential decision making problem into a supervised learning problem. These local guided policies act as response while the indicators act as predictor variables for the machine learning model.

A neural network is used to learn a function that maps from the indicators to the optimal action values using the architecture shown in Figure 4.2(a). Deep neural networks (multi layer perceptrons) can theoretically map a function of infinite dimension and does not over fit when carefully implemented. The Neural network developed in this Chapter fits the model well with good generalization performance.

The model architecture cannot handle the situation when the number of consumers in a μG increase. An improved model should be investigated e.g. recurrent neural network that can handle increase or decrease in the number of consumers. A different model can also be developed based on Chapter 3 and Chapter 4 for learning the hyper parameters using reinforcement learning while still considering the Lyapunov optimization policy for load scheduling.

Research Contributions

This research aims at developing a framework for dynamic scheduling of different types of electrical loads from different types of consumers. Starting with a static approach with full information sharing, this research dwells into the opportunities of using Artificial intelligence with partial information sharing. This research is one of the first research in:

- building a generalized model for dynamic load scheduling while considering different types of electrical loads by different types of consumers
- building a dynamic scheduling model for job shop scheduling while integrating production line loads with non-production line loads
- using forecast values in dynamic load scheduling policy using Lyapunov optimization and present an analysis on selecting the best value of T'
- building a reinforcement learning model for dynamic load scheduling based on optimal values obtained at the end of the planning period

The approach adopted in this research can be generalized to many different problems such as inventory planning and control, vehicle routing, vehicle scheduling, cloud computing and, resource sharing. The idea of using forecasts and guided policy search presented in this research can be used in developing optimization based deep learning model as against the current practice of end-to-end deep learning model without using the information provided by optimization. The idea of solving the problem after realizing the stochastic variables is standard method in operations research's decision making domain. This research integrates this idea of using optimization with neural networks to develop deterministic policies that could make decisions based on real time information.

6.9 Future Research Directions

Dynamic load scheduling problem consists of many interconnected problems e.g. unit commitment, optimal dispatch, infrastructure and capacity planning, and, end user load scheduling. The problems spreads across many disciplines e.g. game theory, security, optimization, simulation, controls, machine learning and infrastructure e.g. battery, storage, new modes of electrical transmission. The current research aims at developing a framework that can take inputs from higher level decisions to provide optimal end-user load scheduling. Thus, this research attempts at solving a small but critical part of the upcoming and environment friendly smart grid infrastructure.

6.9.1 Limitations of the current research

The limitations of this research include:

- Building theoretical models for the selection of T' values for optimal selection of future time slots to consider instead of using sensitivity analysis as depending on the size of the μG, the T' value might change.

- The load scheduling model uses sampling for handling the variance in the different stochastic variables. A theoretical model can be built based on dynamic programming to capture the additional information provided by the variance.

- This research does not consider infrastructure planning as a short time horizon is considered in this research. More sophisticated models can be developed based on this research to solve the load scheduling problem for longer time horizons that can help in the study of infrastructure planning.

- The reinforcement learning problem cannot handle changes in the number of consumers in the μG as with the change in the number of consumers, the input and the output structure changes. This also increases the size of the model and thus increases the model complexity. One direction could be using recurrent neural network that can handle different input sizes.

6.9.2 Research Directions

Future research directions may consider solving the problems posed by limitations of this research. Some other directions include:

- Integrating scheduling with pricing. The current research consider time based pricing but some markets follow Critical Peak Pricing. Also, it is assumed that the pricing is independent of the demand and is based on availability of energy from the renewables and the time based cost of electricity from the Macrogrid.

- Evolutionary computing has been extensively used in scheduling problems. However, they cannot be used in smart grid scenario e.g. μGs in its current form. Modified version of evolutionary computing problems can be considered (example dynamic directions by Ant, or Swarms or learning if - else conditions using Genetic Algorithms)

- Game theoretic approaches have been considered in the context of different dynamic optimization problems with application in smart grid scenario. This research considers that all the consumers are benevolent and participate in automated load scheduling (under mild conditions considered in Chapter 5). However, better incentives models are required to ensure that consumers participate.

- In the reinforcement models discussed in Chapter 6, neural network is used to learn the policy. However, a simple neural network has been used that achieves good results. The results can be improved by making the network more deep. Also, for smaller neural networks, evolutionary computing may be used to find the θ values in the neural network. Population based approaches e.g. particle swarm optimization and ant colony optimization could be used to find the best paths the data flow takes in the neural network.

REFERENCES

REFERENCES

[1] EIA, "Electric Power Annual," Tech. Rep. January, 2010.

[2] "Six maps that show the anatomy of americas vast infrastructure," 2016. [Online]. Available: https://www.washingtonpost.com/graphics/national/maps-of-american-infrastrucure

[3] P. Asmus, "Why microgrids are moving into the mainstream," *IEEE Electrification Magazine*, vol. 2, no. 1, pp. 12–19, 2014.

[4] U. Irfan, "Warming to have severe impacts on peak electricity demand," 2017. [Online]. Available: http://www.eenews.net/climatewire/2017/02/07/stories/1060049626

[5] "Building the digitally powered utility for the future," 2016. [Online]. Available: http://www.ey.com/Publication/vwLUAssets/EY-building-the-digitally-powered-utility-of-the-future

[6] B. P. Asmus, "Are Moving into," *IEEE Electrification Magazine*, no. March, pp. 12–19, 2014.

[7] Kema Inc., "Microgrids Benefits, Models, Barriers and Suggested Policy Initiatives for the Commonwealth of Massachusetts," Tech. Rep., 2014.

[8] J. M. Guerrero, M. Chandorkar, T. L. Lee, and P. C. Loh, "Advanced control architectures for intelligent microgridspart i: Decentralized and hierarchical control," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 4, pp. 1254–1262, 2013.

[9] H. Han, X. Hou, J. Yang, J. Wu, M. Su, and J. M. Guerrero, "Review of Power Sharing Control Strategies for Islanding Operation of AC Microgrids," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 1–16, 2015.

[10] N. Research, 2015. [Online]. Available: http://microgridmedia.com/20x-growth-forecast-in-remote-microgrid-and-nanogrid-market/

[11] 2016. [Online]. Available: http://www.sciencealert.com/costa-rica-has-been-running-on-100-renewable-energy-for-2-months-straight

[12] 2016. [Online]. Available: https://www.engadget.com/2016/11/22/tesla-runs-island-on-solar-power/

[13] EIA, "Electric Power Annual," Tech. Rep. January, 2010.

[14] R. Cespedes and I. Xplore, "Capturing Grid Power 32," Tech. Rep. august, 2009.

[15] H. Ren, W. Gao, and Y. Ruan, "Economic optimization and sensitivity analysis of photovoltaic system in residential buildings," *Renewable Energy*, vol. 34, no. 3, pp. 883 – 889, 2009.

[16] Y. Ru, J. Kleissl, and S. Martinez, "Storage size determination for grid-connected photovoltaic systems," *IEEE Transactions on Sustainable Energy*, vol. 4, no. 1, pp. 68–81, 2013.

[17] A. Gholian, H. Mohsenian-Rad, Y. Hua, and J. Qin, "Optimal Industrial Load Control in Smart Grid: A Case Study for Oil Refineries," *IEEE Transaction on Smart Grid*, vol. 7, no. 5, pp. 2305 – 2316, 2016.

[18] M. Vasirani and S. Ossowski, "A Collaborative Model for Participatory Load Management in the Smart Grid," *1st International Conference on Agreement Technologies*, no. October, pp. 57–70, 2012.

[19] A. H. Mohsenian-Rad and A. Leon-Garcia, "Optimal residential load control with price prediction in real-time electricity pricing environments," *IEEE Transactions on Smart Grid*, vol. 1, no. 2, pp. 120–133, 2010.

[20] M. S. Bhosale and R. A. Pagare, "Residential Load Scheduling using Smart Grid," *International Journal of Science and Research*, vol. 4, no. 7, pp. 2013–2016, 2015.

[21] L. Song, Y. Xiao, and M. Van Der Schaar, "Demand side management in smart grids using a repeated game framework," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 7, pp. 1412–1424, 2014.

[22] H. Kanchev, D. Lu, F. Colas, V. Lazarov, and B. Francois, "Energy management and operational planning of a microgrid with a pv based active generator for smart grid applications," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 10, pp. 4583–4592, 2011.

[23] S. Meters, D. Response, D. Automation, E. V. Charging, and D. Generation, "Smart Grids," vol. 3, no. 0, pp. 1–66, 2012.

[24] A. L.-G. Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, Juri Jatskevich, Robert Schober, "Autonomous Demand Side Management Based on Game-Theoretic Energy Consumption Scheduling for the Future Smart Grid," *IEEE Transaction on Smart Grid*, vol. 1, no. 3, pp. 320–331, 2010.

[25] R. L. Hu, R. Skorupski, R. Entriken, S. Member, and Y. Ye, "A Mathematical Formulation for Optimal Load Shifting of Electricity Demand," *IEEE Transactions on Big Data*, vol. PP, no. 99, pp. 1–18, 2015.

[26] W. Saad, Z. Han, and H. V. Poor, "Coalitional game theory for cooperative micro-grid distribution networks," in *IEEE International Conference on Communications*, 2011, pp. 6–10.

[27] H. S. V. S. K. Nunna and S. Doolla, "Demand response in smart distribution system with multiple microgrids," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1641–1649, 2012.

[28] Z. Huang, T. Zhu, Y. Gu, D. Irwin, A. Mishra, and P. Shenoy, "Minimizing electricity costs by sharing energy in sustainable microgrids," in *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings - BuildSys '14*, 2014, pp. 120–129.

[29] M. S. Wang, Yu and R. M. Nelms, "Distributed online algorithm for optimal real-time energy distribution in the smart grid," vol. 1, no. 1, pp. 70–80, 2014.

[30] Y. Wu, V. K. N. Lau, and D. H. K. Tsang, "Optimal Energy Scheduling for Residential Smart Grid with Centralized Renewable Energy Source," *IEEE Systems Journal*, vol. 8, no. 2, pp. 562–576, 2014.

[31] Siemens, "Siemens press release," 2015. [Online]. Available: http://news.usa.siemens.biz/press-release/siemens-introduces-companys-first-advanced-microgrid-management-software-control-distr

[32] P. Montague, *Reinforcement Learning: An Introduction, by Sutton, R.S. and Barto, A.G.*, 2017, vol. 3, no. 9. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1364661399013315

[33] M.H.Albadi, "Demand response in electricitymarkets: An overview," in *Proc. IEEE 2007 Power Eng. Soc. Gen. Meet.l*, 2007, pp. 1 – 5.

[34] J. C. Fuller, K. P. Schneider, and D. Chassin, "Analysis of residential demand response and double-auction markets," in *IEEE Power and Energy Society General Meeting*, 2011, pp. 1–7.

[35] Z. Sun and L. Li, "Potential capability estimation for real time electricity demand response of sustainable manufacturing systems using Markov Decision Process," *Journal of Cleaner Production*, vol. 65, pp. 184–193, 2013.

[36] J. B. Weiwei Chen, Xing Wang, Jon Petersen, Rajesh Tyagi, "Electric utilities," *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 2309–2318, 2013.

[37] D. Bian, M. Pipattanasomporn, and S. Rahman, "A Human Expert-Based Approach to Electrical Peak Demand Management," *IEEE Transactions on Power Delivery*, vol. 30, no. 3, pp. 1119–1127, 2014.

[38] I. Koutsopoulos and L. Tassiulas, "Control and optimization meet the smart power grid - scheduling of power demands for optimal energy management," *Energy*, p. 9, 2010.

[39] A. Imamura, S. Yamamoto, T. Tazoe, H. Onda, H. Takeshita, S. Okamoto, and N. Yamanaka, "Distributed demand scheduling method to reduce energy cost in smart grid," *2013 IEEE Region 10 Humanitarian Technology Conference*, pp. 148–153, 2013.

[40] M. Vasirani and S. Ossowski, "Smart consumer load balancing: state of the art and an empirical evaluation in the spanish electricity market," *Artificial Intelligence Review*, vol. 39, no. 1, pp. 81–95, 2013.

[41] K. F. L. D. Schlosser, R., "Stochastic optimization of unit com- mitment: a new decomposition framework," *IEEE Transactions on Power Systems*, vol. 11, pp. 21 067 – 1073, 1996.

[42] T. Logenthiran and D. Srinivasan, "Short term generation scheduling of a microgrid," in *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, 2009, pp. 1–6.

[43] Z. M. L. G. Jiang, Ruiwei and Y. Guan, "Two-stage robust power grid optimization problem," *European Journal of Operational Research*, vol. 234, no. 3, pp. 1–34, 2010.

[44] B. Saravanan, S. Das, S. Sikri, and D. P. Kothari, "A solution to the unit commitment problem-a review," *Frontiers in Energy*, vol. 7, no. 2, pp. 223–236, 2013.

[45] W. J. Zheng, Q.P. and A. Liu, "Stochastic optimization for unit commitment a review," *IEEE Transactions on Power Systems*, vol. PP, no. 99, pp. 1–12, 2014.

[46] S. M. Kaplan, *PSmart Grid. Electrical Power Transmission: Background and Policy Issues.* The Capital.Net, Government Series, 2009.

[47] E. Feinberg, "Smart Grid Optimization Smart Grid : What is it ? Smart Grid optimization," Tech. Rep., 2012.

[48] A. Molderink, V. Bakker, M. G. C. Bosman, J. L. Hurink, and G. J. M. Smit, "Management and control of domestic smart grid technology," *IEEE Transactions on Smart Grid*, vol. 1, no. 2, pp. 109 – 119, 2010.

[49] H. Wang and J. Huang, "Bargaining-based energy trading market for interconnected microgrids," *IEEE International Conference on Communications*, vol. 2015-Septe, pp. 776–781, 2015.

[50] J. Qin, S. Member, Y. Chow, S. Member, J. Yang, and S. Member, "Networked Storage Operation Under Uncertainty," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 1–13, 2015.

[51] S. Bakr and S. Cranefield, "Optimizing shiftable appliance schedules across residential neighbourhoods for lower energy costs and fair billing," *CEUR Workshop Proceedings*, vol. 1098, pp. 45–52, 2013.

[52] Z. Baharlouei and M. Hashemi, "Demand Side Management Challenges in Smart Grid: a Review," *Smart Grid Conference*, no. Dlc, pp. 96–101, 2013.

[53] Y. Liu, Y. Zhang, K. Chen, S. Z. Chen, and B. Tang, "Equivalence of Multi-Time Scale Optimization for Home Energy Management Considering User Discomfort Preference," *IEEE Transactions on Smart Grid*, vol. 3053, no. c, pp. 1–1, 2016.

[54] C. Gahm, F. Denz, M. Dirr, and A. Tuma, "Energy-efficient scheduling in manufacturing companies: A review and research framework," *European Journal of Operational Research*, vol. 248, no. 3, pp. 744–757, 2016.

[55] A. Giret, D. Trentesaux, and V. Prabhu, "Sustainability in manufacturing operations scheduling: A state of the art review," *Journal of Manufacturing Systems*, vol. 37, pp. 126–140, 2015.

[56] M. Starke and D. O. E. Eere, *Assessment of Industrial Load for Demand Response across U . S . Regions of the Western Interconnect Prepared by*, 2013, no. September.

[57] M. Starke, D. Letto, N. Alkadi, R. George, B. Johnson, K. Dowling, and S. Khan, "Demand-side response from industrial loads," *2013 NSTI Nanotechnology Conference and Expo*, vol. 2, pp. 758–761, 2013.

[58] G. May, I. Barletta, B. Stahl, and M. Taisch, "Energy management in production: A novel method to develop key performance indicators for improving energy efficiency," *Applied Energy*, vol. 149, pp. 46 – 61, 2015.

[59] E. Zampou, S. Plitsos, A. Karagiannaki, and I. Mourtos, "Towards a framework for energy-aware information systems in manufacturing," *Computers in Industry*, vol. 65, no. 3, pp. 419 – 433, 2014.

[60] F. Shrouf, J. Ordieres, and G. Miragliotta, "Smart factories in industry 4.0: A review of the concept and of energy management approached in production based on the internet of things paradigm," in *2014 IEEE International Conference on Industrial Engineering and Engineering Management*, 2014, pp. 697–701.

[61] P. Mativenga and M. Rajemi, "Calculation of optimum cutting parameters based on minimum energy footprint," *CIRP Annals - Manufacturing Technology*, vol. 60, no. 1, pp. 149–152, jan 2011.

[62] A. Vijayaraghavan and D. Dornfeld, "Automated energy monitoring of machine tools," *CIRP Annals - Manufacturing Technology*, vol. 59, no. 1, pp. 21–24, 2010.

[63] Y. He, B. Liu, X. Zhang, H. Gao, and X. Liu, "A modeling method of task-oriented energy consumption for machining manufacturing system," *Journal of Cleaner Production*, vol. 23, no. 1, pp. 167–174, 2012.

[64] L. B. Z. X. G. H. L. X. He, Yan, "A modeling method of task-oriented energy consumption for machining manufacturing system," *Journal of Cleaner Production*, vol. 23, no. 1, pp. 167–174, 2012.

[65] H. Cao and H. Li, "Simulation-based approach to modeling the carbon emissions dynamic characteristics of manufacturing system considering disturbances," *Journal of Cleaner Production*, vol. 64, pp. 572–580, 2014.

[66] D. Trentesaux and A. Giret, "Go-green manufacturing holons: A step towards sustainable manufacturing operations control," *Manufacturing Letters*, vol. 5, pp. 29 – 33, 2015.

[67] A. Thomas and D. Trentesaux, *Are Intelligent Manufacturing Systems Sustainable?* Cham: Springer International Publishing, 2014, pp. 3–14.

[68] G. B. Davis Meike, Marcello Pellicciari, "Energy efficient use of multirobot production lines in the automotive industry: Detailed system modeling and optimization," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 3, pp. 798 – 809, 2013.

[69] D. Trentesaux and V. Prabhu, *Sustainability in Manufacturing Operations Scheduling: Stakes, Approaches and Trends.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 106–113.

[70] A. Nayak, K. Fang, and S. Lee, "Demand response in flow shop with job due dates using genetic algorithm approach," *Smart and Sustainable Manufacturing Systems*, vol. 1, no. 1, pp. 100–120, 2017.

[71] T. Logenthiran, D. Srinivasan, and T. Z. Shun, "Demand Side Management in Smart Grid Using Heuristic Optimization," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1244–1252, 2012.

[72] J. W. Zhang, H., Zhao, F., Sutherland, "MANUFACTURING SCHEDULING FOR ENERGY COST REDUCTION IN A SMART GRID SCENARIO," in *Proceedings of the ASME 2014 International Manufacturing Science and Engineering Conference MSEC2014 June 9-13, 2014, Detroit, Michigan, USA*, zhang2014j, 2014, pp. 1–10.

[73] T. Majozi, "Heat integration of multipurpose batch plants using a continuous-time framework," *Applied Thermal Engineering*, vol. 26, no. 13, pp. 1369 – 1377, 2006.

[74] L. zdamar and S. I. Birbil, "A hierarchical planning system for energy intensive production environments," *International Journal of Production Economics*, vol. 58, no. 2, pp. 115 – 129, 1999.

[75] G. Mouzon, M. B. Yildirim, and J. Twomey, "Operational methods for minimization of energy consumption of manufacturing equipment," *International Journal of Production Research*, vol. 45, no. 18-19, pp. 4247–4271, 2007.

[76] P. Solding, D. Petku, and N. Mardan, "Using simulation for more sustainable production systems methodologies and case studies," *International Journal of Sustainable Engineering*, vol. 2, no. 2, pp. 111–122, 2009.

[77] Y. He, B. Liu, X. Zhang, H. Gao, and X. Liu, "A modeling method of task-oriented energy consumption for machining manufacturing system," *Journal of Cleaner Production*, vol. 23, no. 1, pp. 167–174, mar 2012.

[78] M. B. Yildirim and G. Mouzon, "Single-machine sustainable production planning to minimize total energy consumption and total completion time using a multiple objective genetic algorithm," *Engineering Management, IEEE Transactions on*, vol. 59, no. 4, pp. 585–597, 2012.

[79] M. Yildirim and G. Mouzon, "A framework to minimise total energy consumption and total tardiness on a single machine," *International Journal of Sustainable Engineering*, vol. 1, no. 2, pp. 1–12, 2014.

[80] E. Seid and T. Majozi, "Optimization of energy and water use in multipurpose batch plants using an improved mathematical formulation," *Chemical Engineering Science*, vol. 111, pp. 335 – 349, 2014.

[81] M. Rager, C. Gahm, and F. Denz, "Energy-oriented scheduling based on evolutionary algorithms," *Computers & Operations Research*, vol. 54, pp. 218 – 231, 2015.

[82] C. Liu, J. Yang, J. Lian, W. Li, S. Evans, and Y. Yin, "Sustainable performance oriented operational decision-making of single machine systems with deterministic product arrival time," *Journal of Cleaner Production*, vol. 85, pp. 318 – 330, 2014.

[83] R. Thry, G. Htreux, M. Agha, A. Hat, and J. L. Lann, "The extended resource task network: a framework for the combined scheduling of batch processes and chp plants," *International Journal of Production Research*, vol. 50, no. 3, pp. 623–646, 2012.

[84] M. H. Agha, R. Thery, G. Hetreux, A. Hait, and J. M. L. Lann, "Integrated production and utility system approach for optimizing industrial unit operations," *Energy*, vol. 35, no. 2, pp. 611 – 627, 2010.

[85] X. Gong, T. D. Pessemier, W. Joseph, and L. Martens, "An energy-cost-aware scheduling methodology for sustainable manufacturing," *Procedia CIRP*, vol. 29, pp. 185 – 190, 2015.

[86] F. Shrouf, J. Ordieres-Mer, A. Garca-Snchez, and M. Ortega-Mier, "Optimizing the production scheduling of a single machine to minimize total energy consumption costs," *Journal of Cleaner Production*, vol. 67, pp. 197 – 207, 2014.

[87] S. Plitsos, P. P. Repoussis, I. Mourtos, and C. D. Tarantilis, "Energy-aware decision support for production scheduling," *Decision Support Systems*, vol. 93, pp. 88–97, 2017.

[88] J. Yusta, F. Torres, and H. Khodr, "Optimal methodology for a machining process scheduling in spot electricity markets," *Energy Conversion and Management*, vol. 51, no. 12, pp. 2647 – 2654, 2010.

[89] H. Yan and L. Fei, "Methods for integrating energy consumption and environmental impact considerations into production operation of machining process," *Chinese Journal of Mechanical Engineering*, vol. 23, pp. 1–8, 2010.

[90] A. Hat and C. Artigues, "A hybrid cp/milp method for scheduling with energy costs," *European Journal of Industrial Engineering*, vol. 5, no. 4, pp. 471–478, 2011.

[91] P. M. Castro, I. Harjunkoski, and I. E. Grossmann, "New continuous-time scheduling formulation for continuous plants under variable electricity cost," *Industrial & Engineering Chemistry Research*, vol. 48, no. 14, pp. 6701–6714, 2009.

[92] L. Tang, P. Che, and J. Liu, "A stochastic production planning problem with nonlinear cost," *Computers & Operations Research*, vol. 39, no. 9, pp. 1977 – 1987, 2012.

[93] Y. yuan TAN, Y. lei HUANG, and S. xin LIU, "Two-stage mathematical programming approach for steelmaking process scheduling under variable electricity price," *Journal of Iron and Steel Research, International*, vol. 20, no. 7, pp. 1 – 8, 2013.

[94] C. V. Le and C. K. Pang, "Fast reactive scheduling to minimize tardiness penalty and energy cost under power consumption uncertainties," *Computers & Industrial Engineering*, vol. 66, no. 2, pp. 406–417, oct 2013.

[95] H.Hadera and I. Harjunkoski, "Continuous-time batch scheduling approach for optimizing electricity consumption cost," *Computer Aided Chemical Engineering*, vol. 32, pp. 403–408, 2013.

[96] D. F. Tobias Kuster, Marco Lutzenberger and S. Albayrak, “Distributed evolutionary optimisation for electricity price responsive manufacturing using multi-agent system technology,” *International Journal on Advances in Intelligent Systems*, vol. 6, no. 2, pp. 27 – 40, 2013.

[97] M. Mashaei and B. Lennartson, “Energy reduction in a pallet-constrained flow shop through on-off control of idle machines,” *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 1, pp. 45–56, 2013.

[98] K.-T. Fang and B. M. Lin, “Parallel-machine scheduling to minimize tardiness penalty and power cost,” *Computers and Industrial Engineering*, vol. 64, no. 1, pp. 224 – 234, 2013.

[99] F. Shrouf, J. Ordieres-Mere, A. Garcia-Sanchez, and M. Ortega-Mier, “Optimizing the production scheduling of a single machine to minimize total energy consumption costs,” *Journal of Cleaner Production*, vol. 67, pp. 197 – 207, 2014.

[100] I. Mattik, P. Amorim, and H.-O. Gnther, “Hierarchical scheduling of continuous casters and hot strip mills in the steel industry: a block planning application,” *International Journal of Production Research*, vol. 52, no. 9, pp. 2576–2591, 2014.

[101] W. Kong, T. Chai, J. Ding, and S. Yang, “Multifurnace optimization in electric smelting plants by load scheduling and control,” *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 3, pp. 850–862, 2014.

[102] U. N. A. Z. F. S. J. W. Fang, Kan, “Scheduling on a single machine under time-of-use electricity tariffs,” *Annals of Operations Research*, vol. 238, no. 1, pp. 199–227, 2016.

[103] J.-Y. Moon and J. Park, “Smart production scheduling with time-dependent and machine-dependent electricity cost by considering distributed energy resources and energy storage,” *International Journal of Production Research*, vol. 52, no. 13, pp. 3922–3939, 2014.

[104] Y. Liu, H. Dong, N. Lohse, S. Petrovic, and N. Gindy, “An investigation into minimising total energy consumption and total weighted tardiness in job shops,” *Journal of Cleaner Production*, vol. 65, pp. 87 – 96, 2014.

[105] M. Georgiadis, L. Papageorgiou, and S. Macchietto, “Optimal cyclic cleaning scheduling in heat exchanger networks under fouling,” *Computers & Chemical Engineering*, vol. 23, Supplement, pp. S203 – S206, 1999.

[106] M. C. Georgiadis and L. G. Papageorgiou, “Optimal scheduling of heat-integrated multipurpose plants under fouling conditions,” *Applied Thermal Engineering*, vol. 21, no. 16, pp. 1675 – 1697, 2001.

[107] M. Ji, J.-Y. Wang, and W.-C. Lee, “Minimizing resource consumption on uniform parallel machines with a bound on makespan,” *Computers & Operations Research*, vol. 40, no. 12, pp. 2970 – 2974, 2013.

[108] K. Li, X. Zhang, J. Y.-T. Leung, and S.-L. Yang, “Parallel machine scheduling problems in green manufacturing industry,” *Journal of Manufacturing Systems*, vol. 38, pp. 98 – 106, 2016.

[109] C.-H. Liu, "Mathematical programming formulations for single-machine scheduling problems while considering renewable energy uncertainty," *International Journal of Production Research*, vol. 54, no. 4, pp. 1122–1133, 2016.

[110] B. J. Zhang, X. L. Luo, X. Z. Chen, and Q. L. Chen, "Coupling process plants and utility systems for site scale steam integration," *Industrial & Engineering Chemistry Research*, vol. 52, no. 41, pp. 14 627–14 636, 2013.

[111] D. Mignon and J. Hermia, "Using batches for modeling and optimizing the brewhouses of an industrial brewery," *Computers & Chemical Engineering*, vol. 17, Supplement 1, pp. S51 – S56, 1993.

[112] C. A. Babu and S. Ashok, "Peak load management in electrolytic process industries," *IEEE Transactions on Power Systems*, vol. 23, no. 2, pp. 399–405, 2008.

[113] K. Nolde and M. Morari, "Electrical load tracking scheduling of a steel plant," *Computers & Chemical Engineering*, vol. 34, no. 11, pp. 1899 – 1903, 2010.

[114] A. Hat and C. Artigues, "On electrical load tracking scheduling for a steel plant," *Computers & Chemical Engineering*, vol. 35, no. 12, pp. 3044 – 3047, 2011.

[115] E. C. Eren and N. Gautam, "Efficient control for a multi-product quasi-batch process via stochastic dynamic programming," *IIE Transactions*, vol. 43, no. 3, pp. 192–206, 2010.

[116] C. Liu, C. Zhao, and Q. Xu, "Integration of electroplating process design and operation for simultaneous productivity maximization, energy saving, and freshwater minimization," *Chemical Engineering Science*, vol. 68, no. 1, pp. 202 – 214, 2012.

[117] A. B. F. D. Zeyi Sun, Lin Li, "Customer-side electricity load management for sustainable manufacturing systems utilizing combined heat and power generation system," *International Journal of Production Economics*, vol. 165, pp. 112 – 119, 2015.

[118] R. Adonyi, J. Romero, L. Puigjaner, and F. Friedler, "Incorporating heat integration in batch process scheduling," *Applied Thermal Engineering*, vol. 23, no. 14, pp. 1743 – 1762, 2003.

[119] K. Nilsson and M. Soderstrom, "Industrial application of production planning with optimal electricity demand," *Applied Energy*, vol. 46, pp. 181–192, 1993.

[120] K. Nilsson, "Industrial production planning with optimal electricity cost," *Energy Conversion and Management*, vol. 34, no. 3, pp. 153 – 158, 1993.

[121] E. Kondili, N. Shah, and C. Pantelides, "Production planning for the rational use of energy in multiproduct continuous plants," *Computers & Chemical Engineering*, vol. 17, Supplement 1, pp. S123 – S128, 1993.

[122] M. P. Brundage, Q. Chang, Y. Li, G. Xiao, and J. Arinez, "Energy efficiency management of an integrated serial production line and hvac system," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 3, pp. 789–797, 2014.

[123] S. Mitra, I. E. Grossmann, J. M. Pinto, and N. Arora, "Optimal production planning under time-sensitive electricity prices for continuous power-intensive processes," *Computers & Chemical Engineering*, vol. 38, pp. 171 – 184, 2012.

[124] Y. Wang and L. Li, "Time-of-use based electricity demand response for sustainable manufacturing systems," *Energy*, vol. 63, pp. 233 – 244, 2013.

[125] N. E.-F. Chudong Tong, Ahmet Palazoglu and X. Yan, "Energy demand management for process systems through production scheduling and control," *AIChE Journal*, vol. 61, no. 11, p. 37563769, 2015.

[126] M. P. Brundage, Q. Chang, Y. Li, J. Arinez, and G. Xiao, "Implementing a real-time, energy-efficient control methodology to maximize manufacturing profits," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 6, pp. 855–866, 2016.

[127] P. M. Castro, I. Harjunkoski, and I. E. Grossmann, "Optimal scheduling of continuous plants with energy constraints," *Computers and Chemical Engineering*, vol. 35, no. 2, pp. 372–387, feb 2011.

[128] D. Mignon and J. Hermia, "Peak utility load reduction in batch processes operated periodically and under uncertainty," *Computers & Chemical Engineering*, vol. 20, no. 3, pp. 249 – 263, 1996.

[129] S. Ashok, "Peak-load management in steel plants," *Applied Energy*, vol. 83, no. 5, pp. 413 – 424, 2006.

[130] K. Fang, N. Uhan, F. Zhao, and J. W. Sutherland, "A new approach to scheduling in manufacturing for power consumption and carbon footprint reduction," *Journal of Manufacturing Systems*, vol. 30, no. 4, pp. 234–240, oct 2011.

[131] K. Fang, N. a. Uhan, F. Zhao, and J. W. Sutherland, "Flow shop scheduling with peak power consumption constraints," *Annals of Operations Research*, vol. 206, no. 1, pp. 115–145, jan 2013.

[132] A. Bruzzone and D. Anghinolfi, "Energy-aware scheduling for improving manufacturing process sustainability: a mathematical model for flexible flow shops," *CIRP Annals*, vol. 61, no. 1, pp. 459–462, 2012.

[133] A. Bego, L. Li, and Z. Sun, "Identification of reservation capacity in critical peak pricing electricity demand response program for sustainable manufacturing systems," *International Journal of Energy Research*, vol. 38, no. 6, pp. 728–736, 2014.

[134] J.-Y. Ding, S. Song, and C. Wu, "Carbon-efficient scheduling of flow shops by multi-objective optimization," *European Journal of Operational Research*, vol. 248, no. 3, pp. 758 – 771, 2016.

[135] C. Boyadjiev, B. Ivanov, N. Vaklieva-Bancheva, C. Pantelides, and N. Shah, "Optimal energy integration in batch antibiotics manufacture," *Computers & Chemical Engineering*, vol. 20, Supplement 1, pp. S31 – S36, 1996, european Symposium on Computer Aided Process Engineering-6.

[136] R. Grau, M. Graells, J. Corominas, A. Espua, and L. Puigjaner, "Global strategy for energy and waste analysis in scheduling and planning of multiproduct batch chemical processes," *Computers & Chemical Engineering*, vol. 20, no. 67, pp. 853 – 868, 1996.

[137] B. Lee and G. Reklaitis, "Optimal scheduling of cyclic batch processes for heat integrationi. basic formulation," *Computers & Chemical Engineering*, vol. 19, no. 8, pp. 883 – 905, 1995.

[138] J. L. Junwen Wang and N. Huang, "Optimal vehicle batching and sequencing to reduce energy consumption and atmospheric emissions in automotive paint shops," *International Journal of Sustainable Manufacturing*, vol. 2, no. 2-3, pp. 1742–7223, 2011.

[139] M. Dai, D. Tang, A. Giret, M. a. Salido, and W. D. Li, "Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm," *Robotics and Computer-Integrated Manufacturing*, vol. 29, no. 5, pp. 418–429, oct 2013.

[140] S. A. Mansouri, E. Aktas, and U. Besikci, "Green scheduling of a two-machine flowshop: Trade-off between makespan and energy consumption," *European Journal of Operational Research*, vol. 248, no. 3, pp. 772 – 788, 2016.

[141] D. Lei, Y. Zheng, and X. Guo, "A shuffled frog-leaping algorithm for flexible job shop scheduling with the consideration of energy consumption," *International Journal of Production Research*, vol. 55, no. 11, pp. 3126–3140, 2017. [Online]. Available: http://dx.doi.org/10.1080/00207543.2016.1262082

[142] H. Zhang, F. Zhao, K. Fang, and J. W. Sutherland, "Energy-conscious flow shop scheduling under time-of-use electricity tariffs," *CIRP Annals - Manufacturing Technology*, vol. 63, no. 1, pp. 37–40, 2014.

[143] E. Capn-Garca, A. D. Bojarski, A. Espua, and L. Puigjaner, "Multiobjective optimization of multiproduct batch plants scheduling under environmental and economic concerns," *AIChE Journal*, vol. 57, no. 10, 2011.

[144] H. Luo, B. Du, G. Q. Huang, H. Chen, and X. Li, "Hybrid flow shop scheduling considering machine electricity consumption cost," *International Journal of Production Economics*, vol. 146, no. 2, pp. 423 – 439, 2013.

[145] O. Adekola, J. D. Stamp, T. Majozi, A. Garg, and S. Bandyopadhyay, "Unified approach for the optimization of energy and water in multipurpose batch plants using a flexible scheduling framework," *Industrial & Engineering Chemistry Research*, vol. 52, no. 25, pp. 8488–8506, 2013.

[146] S. Zanoni, L. Bettoni, and C. H. Glock, "Energy implications in a two-stage production system with controllable production rates," *International Journal of Production Economics*, vol. 149, pp. 164 – 171, 2014.

[147] H. Zhang, F. Zhao, and J. W. Sutherland, "Energy-efficient scheduling of multiple manufacturing factories under real-time electricity pricing," *CIRP Annals - Manufacturing Technology*, vol. 64, no. 1, pp. 41–44, 2015.

[148] Z. F. Zhang, Hao and J. W. Sutherland, "Scheduling of a Single Flow Shop for Minimal Energy Cost Under Real-Time Electricity Pricing," *Journal of Manufacturing Science and Engineering*, vol. 139, no. 1, p. 014502, 2016.

[149] A. Sharma, F. Zhao, and J. W. Sutherland, "Econological scheduling of a manufacturing enterprise operating under a time-of-use electricity tariff," *Journal of Cleaner Production*, vol. 108, Part A, pp. 256 – 270, 2015.

[150] C. Zhang, P. Gu, and P. Jiang, "Low-carbon scheduling and estimating for a flexible job shop based on carbon footprint and carbon efficiency of multi-job processing," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 229, no. 2, pp. 328–342, 2015.

[151] K. Biel, F. Zhao, J. W. Sutherland, C. H. Glock, K. Biel, F. Zhao, J. W. Sutherland, and C. H. Glock, "Flow shop scheduling with grid-integrated onsite wind power using stochastic MILP," *International Journal of Production Research*, vol. 56, no. 5, pp. 2076–2098, 2017.

[152] E.-d. Jiang and L. Wang, "An improved multi-objective evolutionary algorithm based on decomposition for energy- efficient permutation flow shop scheduling problem with sequence-dependent setup time," *International Journal of Production Research*, vol. 0, no. 0, pp. 1–16, 2018. [Online]. Available: https://doi.org/10.1080/00207543.2018.1504251

[153] Y. Zhai, K. Biel, F. Zhao, and J. W. Sutherland, "Dynamic scheduling of a flow shop with on-site wind generation for energy cost reduction under real time electricity pricing," *CIRP Annals - Manufacturing Technology*, vol. 66, no. 1, pp. 41–44, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.cirp.2017.04.099

[154] C. Lu, L. Gao, X. Li, Q. Pan, and Q. Wang, "Energy-efficient permutation flow shop scheduling problem using a hybrid multi-objective backtracking search algorithm," *Journal of Cleaner Production*, vol. 144, pp. 228–238, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.jclepro.2017.01.011

[155] B. Zheng, H. Wu, J. Qin, W. Du, J. Wang, and D. Li, "A simple model clarifies the complicated relationships of complex networks," *arXiv preprint arXiv: . . .*, pp. 1–9, 2012.

[156] C.-H. Liu and D.-H. Huang, "Reduction of power consumption and carbon footprints by applying multi-objective optimisation via genetic algorithms," *International Journal of Production Research*, vol. 52, no. 2, pp. 337–352, 2014.

[157] E. K. Boukas, A. Haurie, and F. Soumis, "Hierarchical approach to steel production scheduling under a global energy constraint," *Annals of Operation Research*, vol. 26, no. 1-4, pp. 289–311, 1991.

[158] C. P. E. Kondili and R. Sargent, "A general algorithm for short term scheduling of batch operations using milp formulation," *Computers & Chemical Engineering*, vol. 17, no. 2, pp. 211 – 227, 1993.

[159] C. Artigues, P. Lopez, and A. Hait, "The Energy Scheduling Problem:Industrial Case Study and constraint propagation techniques," *international journal of production economics*, vol. 143, no. 1, pp. 13–23, 2013.

[160] G. Chen, L. Zhang, J. Arinez, and S. Biller, "Energy-efficient production systems through schedule-based operations," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 1, pp. 27–37, 2013.

[161] C. Pach, T. Berger, Y. Sallez, T. Bonte, E. Adam, and D. Trentesaux, "Reactive and energy-aware scheduling of flexible manufacturing systems using potential fields," *Computers in Industry*, vol. 65, no. 3, pp. 434 – 448, 2014.

[162] Y. He, Y. Li, T. Wu, and J. W. Sutherland, "An energy-responsive optimization method for machine tool selection and operation sequence in flexible machining job shops," *Journal of Cleaner Production*, vol. 87, pp. 245 – 254, 2015.

[163] D. M. Tang, Dunbing, "Energy-efficient approach to minimizing the energy consumption in an extended job-shop scheduling problem," *Chinese Journal of Mechanical Engineering*, vol. 28, no. 5, pp. 1048–1055, 2015.

[164] M. A. Salido, J. Escamilla, A. Giret, and F. Barber, "A genetic algorithm for energy-efficiency in job-shop scheduling," *The International Journal of Advanced Manufacturing Technology*, vol. 85, no. 5, pp. 1303–1314, 2016.

[165] C. K. Pang and C. V. Le, "Optimization of total energy consumption in flexible manufacturing systems using weighted p-timed petri nets and dynamic programming," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 4, pp. 1083–1096, 2014.

[166] R. Zhang and R. Chiong, "Solving the energy-efficient job shop scheduling problem: a multi-objective genetic algorithm with enhanced local search for minimizing the total weighted tardiness and total energy consumption," *Journal of Cleaner Production*, vol. 112, Part 4, pp. 3361 – 3375, 2016.

[167] L. X. G. L.-Z. G. Zhang, Liping, "Dynamic rescheduling in fms that is simultaneously considering energy consumption and schedule efficiency," *The International Journal of Advanced Manufacturing Technology*, vol. 87, no. 5, pp. 1387–1399, 2016.

[168] G. May, B. Stahl, M. Taisch, and V. Prabhu, "Multi-objective genetic algorithm for energy-efficient job shop scheduling," *International Journal of Production Research*, vol. 53, no. 23, pp. 7071–7089, 2015.

[169] M. A. Salido, J. Escamilla, F. Barber, and A. Giret, "Rescheduling in job-shop problems for sustainable manufacturing systems," *Journal of Cleaner Production*, vol. 162, pp. S121–S132, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.jclepro.2016.11.002

[170] M. Dai, D. Tang, Y. Xu, and W. Li, "Energy-aware integrated process planning and scheduling for job shops," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 229, no. 1_suppl, pp. 13–26, 2015.

[171] L. Yin, X. Li, L. Gao, C. Lu, and Z. Zhang, "Energy-efficient job shop scheduling problem with variable spindle speed using a novel multi-objective algorithm," *Advances in Mechanical Engineering*, vol. 9, no. 4, pp. 1–21, 2017.

[172] X. Wu and Y. Sun, "A green scheduling algorithm for flexible job shop with energy-saving measures," *Journal of Cleaner Production*, vol. 172, pp. 3249–3264, 2018.

[173] H. Piroozfard, K. Y. Wong, and W. P. Wong, "Minimizing total carbon footprint and total late work criterion in flexible job shop scheduling by using an improved multi-objective genetic algorithm," *Resources, Conservation and Recycling*, vol. 128, pp. 267–283, 2018.

[174] J.-Y. Moon and J. Park, "Smart production scheduling with time-dependent and machine-dependent electricity cost by considering distributed energy resources and energy storage," *International Journal of Production Research*, vol. 52, no. 13, pp. 1–18, dec 2013.

[175] C. Garcia-Santiago, J. D. Se, C. Upton, F. Quilligan, S. Gil-Lopez, and S. Salcedo-Sanz, "A random-key encoded harmony search approach for energy-efficient production scheduling with shared resources," *Engineering Optimization*, vol. 47, no. 11, pp. 1481–1496, 2015.

[176] R. Z. R. C. J. Y. Ding, S. Song and C. Wu, "Parallel machine scheduling under time-of-use electricity prices: New models and optimization approaches," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 1138–1154, 2016.

[177] T. Stock and G. Seliger, "Multi-objective shop floor scheduling using monitored energy data," *Procedia CIRP*, vol. 26, pp. 510 – 515, 2015.

[178] H. Mokhtari and A. Hasani, "An energy-efficient multi-objective optimization for flexible job-shop scheduling problem," *Computers and Chemical Engineering*, vol. 104, pp. 339–352, 2017.

[179] Y. Zhang, J. Wang, and Y. Liu, "Game theory based real-time multi-objective flexible job shop scheduling considering environmental impact," *Journal of Cleaner Production*, vol. 167, pp. 665–679, 2018.

[180] D. Lei and X. Guo, "An effective neighborhood search for scheduling in dual-resource constrained interval job shop with environmental objective," *International Journal of Production Economics*, vol. 159, pp. 296 – 303, 2015.

[181] C. Liu, F. Dang, W. Li, J. Lian, S. Evans, and Y. Yin, "Production planning of multi-stage multi-option seru production systems with sustainable measures," *Journal of Cleaner Production*, vol. 105, pp. 285 – 299, 2015.

[182] G. Gong, Q. Deng, W. Gong, Xuran Liu, and Q. Ren, "A new double flexible job-shop scheduling problem integrating processing time, green production, and human factor indicators," *Journal of Cleaner Production*, vol. 174, no. 23, pp. 560–576, 2018.

[183] L. Yin, X. Li, L. Gao, C. Lu, and Z. Zhang, "A novel mathematical model and multi-objective method for the low-carbon flexible job shop scheduling problem," *Sustainable Computing: Informatics and Systems*, vol. 13, pp. 15–30, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.suscom.2016.11.002

[184] G. May, B. Stahl, M. Taisch, and V. Prabhu, "Multi-objective genetic algorithm for energy-efficient job shop scheduling," *International Journal of Production Research*, vol. 53, no. 23, pp. 7071–7089, 2015.

[185] Z. Sun, L. Li, M. Fernandez, and J. Wang, "Inventory control for peak electricity demand reduction of manufacturing systems considering the tradeoff between production loss and energy savings," *Journal of Cleaner Production*, vol. 82, pp. 84 – 93, 2014.

[186] T. L. Urban and W.-C. Chiang, "Designing energy-efficient serial production lines: The unpaced synchronous line-balancing problem," *European Journal of Operational Research*, vol. 248, no. 3, pp. 789 – 801, 2016.

[187] J. M. Nilakantan, G. Q. Huang, and S. Ponnambalam, "An investigation on minimizing cycle time and total energy consumption in robotic assembly line systems," *Journal of Cleaner Production*, vol. 90, pp. 311 – 325, 2015.

[188] Z. Sun and L. Li, "Potential capability estimation for real time electricity demand response of sustainable manufacturing systems using markov decision process," *Journal of Cleaner Production*, vol. 65, pp. 184 – 193, 2014.

[189] A. Khodaei and M. Shahidehpour, "Microgrid-based co-optimization of generation and transmission planning in power systems," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1582–1590, 2013.

[190] S. Lakshminarayana, T. Q. S. Quek, and H. V. Poor, "Cooperation and storage tradeoffs in power grids with renewable energy resources," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 7, pp. 1386–1397, 2014.

[191] R. Rigo-Mariani, B. Sareni, and X. Roboam, "Integrated Optimal Design of a Smart Microgrid With Storage," *IEEE Transactions on Smart Grid*, pp. 1–9, 2015.

[192] H. Zhang, F. Zhao, and J. W. Sutherland, "Manufacturing scheduling for reduced energy cost in a smart grid scenario," *Reengineering Manufacturing for Sustainability*, pp. 183–189, 2013.

[193] P. Finn and C. Fitzpatrick, "Demand side management of industrial electricity consumption: Promoting the use of renewable energy through real-time pricing," *Applied Energy*, vol. 113, pp. 11 – 21, 2014.

[194] Y.-C. Choi and P. Xirouchakis, "A holistic production planning approach in a reconfigurable manufacturing system with energy consumption and environmental effects," *International Journal of Computer Integrated Manufacturing*, vol. 28, no. 4, pp. 379–394, 2015.

[195] F. Z. Hao Zhang and J. W. Sutherland, "Manufacturing scheduling of collaborative factories for energy cost reduction." 43rd SME North American Manufacturing Research Conference (NAMRC), Charlotte, 2015, pp. 273–278.

[196] L. L. Zeyi Sun and F. Dababneh, "Plant-level electricity demand response for combined manufacturing system and heating, venting, and air-conditioning system," *Journal of Cleaner Production*, vol. 135, pp. 1650 – 1657, 2016.

[197] T. Stock and G. Seliger, "Multi-objective shop floor scheduling using monitored energy data," *Procedia CIRP*, vol. 26, pp. 510–515, 2015.

[198] M. J. Neely and L. Huang, "Dynamic product assembly and inventory control for maximum profit," *Proceedings of the IEEE Conference on Decision and Control*, pp. 2805–2812, 2010.

[199] S. T. Maguluri, R. Srikant, and L. Ying, "Stochastic models of load balancing and scheduling in cloud computing clusters," *Proceedings - IEEE INFOCOM*, pp. 702–710, 2012.

[200] M. J. Neely, "A lyapunov optimization approach to repeated stochastic games," in *51st Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL*, 2013, pp. 1082–1089.

[201] ——, *Stochastic Network Optimization with Application to Communication and Queueing Systems*, 2010, vol. 3, no. 1.

[202] T. Chen, Q. Ling, and G. B. Giannakis, "Learn-and-Adapt Stochastic Dual Gradients for Network Resource Allocation," pp. 1–14, 2017.

[203] S. T. Neely, M.J. and G. Dimakis, "Efficient algorithms for renewable energy allocation to delay tolerant consumers," *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, no. 1, pp. 1–10, 2010.

[204] S. Salinas, M. Li, P. Li, and Y. Fu, "Dynamic Energy Management for the Smart Grid With Distributed Energy Resources," *Smart Grid, IEEE Transactions on*, vol. 4, no. 4, pp. 2139–2151, 2013.

[205] Y. Huang, S. Mao, and R. M. Nelms, "Adaptive electricity scheduling in microgrids," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 270–281, 2014.

[206] S. Lakshminarayana, T. Q. S. Quek, and H. V. Poor, "Combining cooperation and storage for the integration of renewable energy in smart grids," in *Proceedings - IEEE INFOCOM*, 2014, pp. 622–627.

[207] T. Li and M. Dong, "Real-time energy storage management: Finite-time horizon approach," *2014 IEEE International Conference on Smart Grid Communications, SmartGridComm 2014*, pp. 115–120, 2015.

[208] W. Shi, N. Li, C. C. Chu, and R. Gadh, "Real-Time Energy Management in Microgrids," *IEEE Transactions on Smart Grid*, vol. 8, no. 1, pp. 228–238, 2017.

[209] M. Yu and S. H. Hong, "A Real-Time Demand-Response Algorithm for Smart Grids: A Stackelberg Game Approach," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 879–888, 2015.

[210] M. Moghaddam and S. Y. Nof, "Combined demand and capacity sharing with best matching decisions in enterprise collaboration," *International Journal of Production Economics*, vol. 148, pp. 93–109, 2014.

[211] H. Seok and S. Y. Nof, "Collaborative capacity sharing among manufacturers on the same supply network horizontal layer for sustainable and balanced returns," *International Journal of Production Research*, vol. 52, no. 6, pp. 1622–1643, 2014.

[212] S. W. Yoon and S. Y. Nof, "Demand and capacity sharing decisions and protocols in a collaborative network of enterprises," *Decision Support Systems*, vol. 49, no. 4, pp. 442–450, 2010.

[213] H. Seok and S. Nof, "Collaborative capacity sharing among manufacturers on the same supply network horizontal layer for sustainable and balanced returns," *International Journal of Production Research*, vol. 52, no. January 2015, pp. 1622–1643, 2014.

[214] S. W. Yoon and S. Y. Nof, "Affiliation/dissociation decision models in demand and capacity sharing collaborative network," *International Journal of Production Economics*, vol. 130, no. 2, pp. 135–143, 2011.

[215] M. Moghaddam and S. Y. Nof, "Real-time optimization and control mechanisms for collaborative demand and capacity sharing," *International Journal of Production Economics*, vol. 171, Part 4, pp. 495 – 506, 2016.

[216] M. Mohsen and S. Y. Nof, *Frontiers in Best Matching.* Springer International Publishing, 2017, pp. 221–228.

[217] S. L. Ok C., P. Mitra and S. Kumara, "Maximum energy welfare routing in wireless sensor networks," in *Proceedings of Networking*, vol. 4479, 2007, p. 203214.

[218] L. S. M. P. Ok, C. and S. Kumara, "Distributed routing in wireless sensor networks using energy welfare metric," *Information Science*, vol. 180, no. 9, pp. 1656–1670, 2010.

[219] P. T. Yang and S. Lee, "A distributed reclustering hierarchy routing protocol using social welfare in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 8, no. 4, p. 681026, 2012.

[220] S. Lee, "The role of preparedness in ambulance dispatching," *Journal of the Operational Research Society*, vol. 62, p. 1888, 2011.

[221] M. H. Kim, S. P. Kim, and S. Lee, "Social-welfare based task allocation for multi-robot systems with resource constraints," *Computers and Industrial Engineering*, vol. 63, no. 4, pp. 994–1002, 2012.

[222] J. Kim, H. Chae, S.-H. Yook, and Y. Kim, "Spatial evolutionary public goods game on complete graph and dense complex networks," *Scientific Reports*, vol. 5, p. 9381, 2015.

[223] C. Ok, S. Lee, and S. Kumara, "Group preference modeling for intelligent shared environments: Social welfare beyond the sum," *Information Sciences*, vol. 278, pp. 588–598, 2014.

[224] A. Nayak, R. Levalle, S. Lee, and S. Y. Nof, "Resource sharing in cyber-physical systems: modeling framework and case studies," *International Journal of Production Research*, vol. 7543, no. 3, pp. 1–15, 2016.

[225] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-End Training of Deep Visuomotor Policies," vol. 17, pp. 1–40, 2015. [Online]. Available: http://arxiv.org/abs/1504.00702

[226] M. Aydin and E. Öztemel, "Dynamic job-shop scheduling using reinforcement learning agents," *Robotics and Autonomous Systems*, vol. 33, no. 2-3, pp. 169–178, 2000.

[227] R. Glaubius, T. Tidwell, C. Gill, and W. D. Smart, "Real-Time Scheduling via Reinforcement Learning," in *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence (UAI)*, 2010, pp. 1–9.

[228] E. C. Kara, M. Berges, B. Krogh, and S. Kar, "Using smart devices for system-level management and control in the smart grid: A reinforcement learning framework," in *2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm)*, 2012, pp. 85–90. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6485964

[229] D. Li and S. K. Jayaweera, "Reinforcement learning aided smart-home decision-making in an interactive smart grid," in *2014 IEEE Green Energy and Systems Conference, IGESC 2014*, 2015, pp. 1–6.

[230] B.-G. Kim, Y. Zhang, M. van der Schaar, and J.-W. Lee, "Dynamic pricing for smart grid with reinforcement learning," in *2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2014, pp. 640–645. [Online]. Available: http://ieeexplore.ieee.org/document/6849306/

[231] Y. Zhang and M. van der Schaar, "Structure-aware stochastic load management in smart grids," in *INFOCOM, 2014 Proceedings IEEE*, no. Dlc, 2014, pp. 2643–2651.

[232] L. Qing, Z. Xiaoshun, P. Zhenning, T. Min, G. L. Yongkun, Y. Tao, L. Qianjin, and Y. Feng, "Decentralized reinforcement learning collaborative consensus algorithm for generation dispatch in virtual generation tribe," in *2015 IEEE,Electronic and Automation Control Conference (IAEAC)*, 2015, pp. 5–9.

[233] T. Yu, H. Z. Wang, B. Zhou, K. W. Chan, and J. Tang, "Multi-Agent Correlated Equilibrium Q(λ) Learning for Coordinated Smart Generation Control of Interconnected Power Grids," *IEEE Transactions on Power Systems*, vol. 30, no. 4, pp. 1669–1679, 2015.

[234] R. T. Kuate, M. Chli, and H. H. Wang, "Optimising market share and profit margin: SMDP-based tariff pricing under the smart grid paradigm," *IEEE PES Innovative Smart Grid Technologies Conference Europe*, vol. 3, no. January, pp. 1–6, 2015.

[235] Y. Wang, S. Member, X. Lin, S. Member, M. Pedram, and A. Integrating, "A Near-Optimal Model-Based Control Algorithm for Households Equipped With Residential Photovoltaic Power Generation and Energy Storage Systems," *IEEE Transaction on Sustainable Energy*, vol. 7, no. 1, pp. 1–10, 2015.

[236] M. Rayati, A. Sheikhi, and A. M. Ranjbar, "Applying reinforcement learning method to optimize an Energy Hub operation in the smart grid," *2015 IEEE Power and Energy Society Innovative Smart Grid Technologies Conference, ISGT 2015*, 2015.

[237] X. Pan, "An Approach of Reinforcement Learning Based Lighting Control for Demand Response," in *PCIM Europe 2016, 10 12 May 2016, Nuremberg, Germany An*, no. May, 2016, pp. 10–12.

[238] D. Jay and K. S. Swarupt, "Demand Response based Automatic Generation Control in Smart-Grid Deregulated Market," in *2016 IEEE 6th International Conference on Power Systems (ICPS)*, 2016, p. 8.

[239] S. Ahmed and F. Bouffard, "Building Load Management Clusters Using Reinforcement Learning," in *2017 8th IEEE Annual, Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2017, pp. 372–377.

[240] F. Ruelens, B. J. Claessens, S. Vandael, B. De Schutter, R. Babuska, and R. Belmans, "Residential Demand Response of Thermostatically Controlled Loads Using Batch Reinforcement Learning," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2149–2159, 2017.

[241] W. Chen, Y. Xu, and X. Wu, "Deep Reinforcement Learning for Multi-Resource Multi-Machine Job Scheduling," in *arXiv preprint arXiv:1711.07440v1*, 2017, pp. 1–2. [Online]. Available: http://arxiv.org/abs/1711.07440v1%0Apapers3://publication/uuid/14FD8391-86DA-4210-A587-B2775CB60EF2

[242] Z. D. Asl, V. Derhami, and M. Yazdian-dehkordi, "A new Approach on Multi-Agent Multi-Objective Reinforcement Learning based on agents ' preferences," in *2017 Artificial Intelligence and Signal Processing Conference (AISP)*, 2017, pp. 75–79.

[243] Jasmin E.A., T. Ahamed, and Remani T., "A function approximation approach to Reinforcement Learning for solving unit commitment problem with Photo voltaic sources," in *2016 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*, 2016, pp. 1–6. [Online]. Available: http://ieeexplore.ieee.org/document/7914428/

[244] X. Zhang, T. Bao, T. Yu, B. Yang, and C. Han, "Deep transfer Q-learning with virtual leader-follower for supply-demand Stackelberg game of smart grid," *Energy*, vol. 133, pp. 348–365, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.energy.2017.05.114

[245] S. Bahrami, V. W. Wong, and J. Huang, "An Online Learning Algorithm for Demand Response in Smart Grid," *IEEE Transactions on Smart Grid*, vol. 3053, no. c, pp. 1–1, 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7849144/

[246] S. Levine and V. Koltun, "Variational policy search via trajectory optimization," in *Advances in Neural Information . . .* , 2013, pp. 1–9. [Online]. Available: http://machinelearning.wustl.edu/mlpapers/paper_files/NIPS2013_5178.pdf

[247] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic Policy Gradient Algorithms," *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 387–395, 2014.

[248] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in *ICLR 2016 CONTINUOUS*, 2015. [Online]. Available: http://arxiv.org/abs/1509.02971

[249] R. Lioutikov, A. Paraschos, J. Peters, and G. Neumann, "Sample-Based Information-Theoretic Stochastic Optimal Control," *Proceedings of 2014 IEEE International Conference on Robotics and Automation*, pp. 3896–3902, 2014.

[250] "Hourly electricity prices," 2016. [Online]. Available: https://www2.ameren.com/RetailEnergy/RtpDownload

[251] "Data catalog," 2016. [Online]. Available: https://catalog.data.gov/dataset/commercial-and-residential-hourly-load-profiles-for-all-tmy3-locations-in-the-united-state-1d21c

[252] "Generation from renewables," 2016. [Online]. Available: http://www.ercot.com/gridinfo/generation

[253] A. Nayak, "Gihub link to the codes," 2018. [Online]. Available: https://www.github.com/ashutoshnayakIE

[254] E. foundation, "Eclipse ide," 2014. [Online]. Available: https://www.eclipse.org/

[255] jet Brains, "Pycharm ide," 2015. [Online]. Available: https://www.jetbrains.com/pycharm/

[256] Gurobi, "Gurobi optimization solver," 2015. [Online]. Available: http://www.gurobi.com

[257] Google, "Tensorflow," 2017. [Online]. Available: https://www.tensorflow.org/

[258] Ameren, "Hourly electricity prices," 2016. [Online]. Available: https://www2.ameren.com/RetailEnergy/RtpDownload

[259] S. Lakshminarayana, T. Q. S. Quek, and H. V. Poor, "Combining cooperation and storage for the integration of renewable energy in smart grids," in *Proceedings - IEEE INFOCOM*, 2014, pp. 622–627.

[260] T. Jiang, Y. Cao, L. Yu, and Z. Wang, "Load shaping strategy based on energy storage and dynamic pricing in smart grid," *IEEE Transactions on Smart Grid*, vol. 5, no. 6, pp. 2868–2876, Nov 2014.

[261] Z. Bao, W. Qiu, L. Wu, F. Zhai, W. Xu, B. Li, and Z. Li, "Optimal Multi-Timescale Demand Side Scheduling Considering Dynamic Scenarios of Electricity Demand," *IEEE Transactions on Smart Grid*, vol. 3053, no. c, pp. 1–1, 2018.

[262] L. Song, Y. Xiao, and M. van der Schaar, "Demand side management in smart grids using a repeated game framework," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 7, pp. 1412–1424, July 2014.

[263] Y. Z. Li, P. Wang, H. B. Gooi, J. Ye, and L. Wu, "Multi-Objective Optimal Dispatch of Microgrid under Uncertainties via Interval Optimization," *IEEE Transactions on Smart Grid*, vol. 3053, no. c, 2017.

[264] S. Ho, M. Xie, and T. Goh, "A comparative study of neural network and box-jenkins arima modeling in time series prediction," *Computers & Industrial Engineering*, vol. 42, no. 2-4, pp. 371–375, 2002.

[265] E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," *CoRR*, vol. abs/1012.2599, 2010.

[266] R. Martinez-Cantin, "Bayesopt: A bayesian optimization library for nonlinear optimization, experimental design and bandits," *Journal of Machine Learning Research*, vol. 15, pp. 1–5, 2014.

[267] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, 2015.

[268] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2015. [Online]. Available: http://arxiv.org/abs/1509.02971

APPENDIX

APPENDIX

Using Raking and selection for selecting the best value for T'

A bayesian approach for the selection of T' is described in this Section. The method is known as Ranking and Selection in the Simulation community. It is known as Thompson's Sampling or multi-lever slot machine optimization in dynamic programming community. The method has asymptotic guarantees but it cannot guarantee optimal values for different options but obtains the "best" value. The best alternative or choice is selected based on the measure of output. In this chapter, total electricity cost is considered as a measure. The sensitivity is performed for $g = \{1, 2, ..., 24\}$ values of T'. Since, the experiments are simulated based on random variables, the outcome from the experiments is also a random variable. The objective is to sequentially select a value of g such that the expected value of χ is maximized where χ is defined in Equation 6.21 and $\mu(g_i)$ is the utility from g_i where utility is defined as $u(g) = -cost$. The method is flexible as different objective functions e.g. throughput, customer satisfaction may also be considered for the cost function.

$$\chi = max_{g=1,2,...,G}\, \mu(g) \tag{6.21}$$

Since μ_i is random, it has a sampling distribution. Assuming that the sampling distribution is normal, μ_i $N(\mu_o(g), \sigma_o^2(g))$ where are the parameters of the prior distribution. Since the prior distribution is not known, the experiment is run 10 times to estimate $\mu_o(g)$, $\sigma_o^2(g)$ and, $\lambda(g)$ where $\lambda(g)$ is the estimation for the population variance. The sequential procedure for Bayesian ranking and selection is described below where the experiments are run NE times with optimal allocation of simulation effort. Each simulation run is denoted with n.

- The experiments are run 10 times for every value of g to estimate $\mu_o(g)$ and $\sigma_o^2(g)$
- The choice of g to sample is obtained from the function shown as:

$$argmax_{g=1,2,\ldots,G}\, \tilde{\sigma}_n(g) f\left(- \frac{|\triangle_n(g)|}{\tilde{\sigma}_n(g)} \right) \tag{6.22}$$

  where $f(z) = z\phi(z) + \psi(z)$. $\phi(z)$ is the normal CDF and $\psi(z)$ is the normal PDF. $\tilde{\sigma}_n(g)$ and $\triangle_n(g)$ are calculated as shown in Equation 6.23 and Equation 6.24 respectively.

$$\tilde{\sigma}_n(g) = \sqrt{\sigma_n^2(g) - \frac{1}{\sigma_n^{-2}(g) + \lambda(g)}} \tag{6.23}$$

$$\triangle_n(g) = \mu_n(g) - max_{g' \neq g}\, \mu_n(g') \tag{6.24}$$

  $\mu_n(g)$ and $\sigma_n(g)$ are obtained from Equation 6.25 and Equation 6.26 respectively where $y(g)$ is the new observation for choice g..

$$\mu_n(g) = \frac{\sigma_{n-1}^{-2}(g)\mu_{n-1}(g) + \lambda^{-1}(g) y_n(g)}{\sigma_o^{-2}(g) + \lambda^{-1}(g)} \tag{6.25}$$

$$\sigma_n^2(g) = \frac{1}{\sigma_o^{-2}(g) + \lambda^{-1}(g)} \tag{6.26}$$

- After NE runs, select the value of g with highest expected value for Equation 6.22

The best values of T' for different test cases is shown in Table 5.2. The results are obtained for storage option 1 and load factor 0.25. The optimal values of T' depends on the structure of the problem and distribution of the random variables. The total number of simulation runs (or simulation effort) is kept at 5000 over which the allocation of simulation effort is made. In the sequential approach, in each step, only one choice is selected and a random sample is sampled for that choice. Bayesian approach for ranking and selection provides a statistical method to obtain the best choice that is expected to perform good on an average. However, it may not perform well when considering worst case measures.

Theorem 1: The virtual queue Q_i in Chapter 3 is mean rate stable

Proof A queue $Q_i[t]$ is mean rate stable if Equation 6.27 is satisfied.

$$\lim_{|T|\to\infty} \frac{1}{|T|} E\big[Q_i[t]\big] = 0 \tag{6.27}$$

Using the drift function $\triangle[t]$, taking an expectation over $Q[t]$ and summing it over all t, $\triangle[t]$ is converted to the form as shown in Equation 6.28.

$$\omega[t] - \omega[0] \leq C|T| + -\sum_{t\in T} Q_i[t] \sum_{d\in D} L^{sl}_{i,d} w^{sl}_{i,d} x^{sl}_{i,d} \tag{6.28}$$

Let $Q_i[0] = 0$, $\forall i \in I$, hence $\omega[0]$. Right hand side of Equation 6.28 consist of two parts where $C|T|$ is constant. In the second part, $Q_i[t] \geq 0$ and $x^{sl}_{i,d} \geq 0$. Hence, Equation 6.28 is simplified to Equation 6.29.

$$\sum_{i\in I} E\big[Q_i[t]^2\big] \leq C|T| \tag{6.29}$$

As $E[a^2] \geq E[a]^2$, Equation 6.30 is derived from Equation 6.29.

$$E[Q_i[t]] \leq \sqrt{C|T|} \quad \forall i \in I \tag{6.30}$$

Dividing both sides by $|T|$ makes $\lim_{|T|\to\infty} \sqrt{\frac{C}{|T|}} = 0$, that concludes the proof as the desired result shown in Equation 6.27 is obtained.

Theorem 2: Penalty Term in $\nabla[t]$ is bounded by $\frac{1}{V}$ and queue term by V

Proof The drift+penalty inequality term is given as $\nabla[t] \leq C - \sum_{i \in I} \delta_i E\Big[Q_i[t]\Big] + V\Big[X[t]p_1[t] + G[t]p_2[t]\Big]$. Let $M[t] = X[t]p_1[t] + G[t]X_2[t]$, M^* be the optimal value of $E[M]$ and, M_{min} be the lower bound on the M. Summing $\nabla[t]$ over all t, Equation 6.31 is obtained.

$$E\big[\omega[Q[t]]\big] - E\big[\omega[Q[0]]\big] + VM \leq C|T| + VM^* - \sum_{i \in I}\sum_{t \in T} \delta_i E\big[Q_i[t]\big] \tag{6.31}$$

Using Theorem 1 and applying the limit on both sides, Equation 6.32 is obtained that shows the penalty term is bounded by $\frac{1}{V}$.

$$\lim_{|T| \to \infty} \frac{1}{|T|} \sum_{t \in T} M[t] \leq M^* + \frac{D}{V} \tag{6.32}$$

Using Theorem 1, rearranging Equation 6.31 and putting the limits, Equation 6.33 is constructed that shows direct proportionality between queue term and V. Equation 6.34 concludes the proof.

$$\lim_{|T| \to \infty} \frac{1}{|T|} \sum_{i \in I} \delta_i E\Big[Q_i[t]\Big] \leq C|T| + VM^* - V \lim_{|T| \to \infty} \frac{1}{|T|} E\Big[M[t]\Big] \tag{6.33}$$

$$\lim_{|T| \to \infty} \frac{1}{|T|} E\Big[Q_i[t]\Big] \leq \frac{C|T| + V\big(M^* - M_{min}\big)}{\delta_i} \quad \forall i \in I \tag{6.34}$$

Theorem 3: The virtual queues Q_i^{CCE} and Q_i^P are mean rate stable

Proof Using the proofs similarly as in Theorem 1, Equation 6.35 and Equation 6.36 are obtained.

$$\begin{aligned}\omega^{CCE}[t] - \omega^{CCE}[0] =& |T|\Bigg(\sum_{i\in I}\big(-p_2[t](r_i[t]-m_i[t]) + \phi K_i(\frac{1}{m_i[t]} - \frac{1}{r_i[t]})\big)^2\Bigg) \\ &- \sum_{i\in I} Q_i^{CCE}[t]\Bigg(-p_2[t](r_i[t]-m_i[t]) + \phi K_i(\frac{1}{m_i[t]} - \frac{1}{r_i[t]})\Bigg)\end{aligned} \tag{6.35}$$

$$\begin{aligned}\omega^{P}[t] - \omega^{P}[0] =& |T|\Bigg(\sum_{i\in I}\big(-p_2[t]r_i[t] + pm_i[t] + \phi K_i(\frac{1}{m_i[t]} - \frac{1}{r_i[t]})\big)^2\Bigg) \\ &- \sum_{i\in I} Q_i^{CCE}[t]\Bigg(-p_2[t]r_i[t] - pm_i[t] + \phi K_i(\frac{1}{m_i[t]} - \frac{1}{r_i[t]})\Bigg)\end{aligned} \tag{6.36}$$

Taking out the constants and replacing them by C, provides us with the same form as Equation 6.29 that can be used to show the mean rate stability of the virtual queues Q_i^{CCE} and Q_i^P. It is shown in Equation 6.37 and Equation 6.38 respectively. The constants for Equation 6.37 and Equation 6.38 are shown in Equation 6.39 and Equation 6.40. Note that the constant are not exactly constants but are bounded from above.

$$\omega^{CCE}[t] - \omega^{CCE}[0] = C_1 - \sum_{i\in I} Q_i^{CCE}\Bigg(p_2[t]m_i[t] + \frac{\phi K_i}{m_i[t]}\Bigg) \tag{6.37}$$

$$\omega^{P}[t] - \omega^{P}[0] = C_2 - \sum_{i\in I} Q_i^{CCE}\Bigg(pm_i[t] + \frac{\phi K_i}{m_i[t]}\Bigg) \tag{6.38}$$

$$\begin{aligned}C_1 =& |T|\Bigg(\sum_{i\in I}\big(-p_2[t](r_i[t]-m_i[t]) + \phi K_i(\frac{1}{m_i[t]} - \frac{1}{r_i[t]})\big)^2\Bigg) \\ &- \sum_{i\in I} Q_i^{CCE}[t]\Bigg(\frac{\phi K_i}{r_i[t]} + p_2[t]m_i[t]\Bigg)\end{aligned} \tag{6.39}$$

$$\begin{aligned}C_2 =& |T|\Bigg(\sum_{i\in I}\big(-p_2[t]r_i[t] + pm_i[t] + \phi K_i(\frac{1}{m_i[t]} - \frac{1}{r_i[t]})\big)^2\Bigg) \\ &- \sum_{i\in I} Q_i^{P}[t]\Bigg(\frac{\phi K_i}{r_i[t]} + pm_i[t]\Bigg)\end{aligned} \tag{6.40}$$

To make the model simple, linear and convex, the terms $Q_i^{CCE}\frac{1}{m_i[t]}$ and $Q_i^{P}\frac{1}{m_i[t]}$ are not included in Equation 6.37 and Equation 6.38. These terms are bounded as $m_i[t] \geq L_i^{ns}[t]$, thus the term does not go to ∞. Also, K_i is bounded as it is the total electricity requested from a player. Thus, the drift for the player CCE and player participation is given by Equation 6.41 and Equation 6.42 respectively. The loss of these terms can also be compensated by selecting the best value of T_H and V.

$$\omega^{CCE}[t] - \omega^{CCE}[0] = C_1 - \sum_{i \in I} Q_i^{CCE}\Big(p_2[t]m_i[t]\Big) \tag{6.41}$$

$$\omega^{P}[t] - \omega^{P}[0] = C_2 - \sum_{i \in I} Q_i^{CCE}\Big(pm_i[t]\Big) \tag{6.42}$$

VITA

VITA

ASHUTOSH NAYAK

N Salisbury Street, West Lafayette, IN, 47906 — 765 409 6516

EDUCATION

PhD, School of Industrial Engineering, Purdue University	2014 - 2018
M.Tech, Industrial Engineering, Indian Institute of Technology, Kharagpur	2013 - 2014
B.Tech, Industrial Engineering, Indian Institute of Technology, Kharagpur	2009 - 2013

WORK EXPERIENCE

Postdoctoral fellow, University of California, Davis	Dec 2018 - Aug 2020
Research Assistant, INMaC	Aug 2017 - Dec 2018
Intern, General Electric Global Research Center, Niskayuna, NY	May 2016 - Aug 2016
Research Assistant, General Electric	Apr 2015 - May 2017
Teaching Assistant, Purdue University	Aug 2014 - May 2015

RESEARCH INTEREST

Dynamic decision making based on historical aggregation, real time information and forecast

Integrating machine learning with operations research in allied research application

Integrating optimization with reinforcement learning for different application

Understanding customer behavior and designing optimal marketing policies

Optimizing multi-objective manufacturing scheduling problem

ASHUTOSH NAYAK

N Salisbury Street, West Lafayette, IN, 47906 — 765 409 6516

JOURNAL PUBLICATIONS

- Using Reinforcement Learning for Dynamic Load Scheduling in Smart Grid (*)
- Nayak, A. and Lee, S. (2018) Optimal Fermenter Scheduling For Electricity Cost Minimization At A Chemical Production Facility (**Journal of Cleaner Productions**)
- Nayak, A. and Lee, S (2017) Storage Sizing and Optimal Load Scheduling for Cooperative Consumers in μG with Different Load Types, IISE Transactions, 0 (0), 1-9
- Nayak, A., Fang, K. and Lee, S (2017) An Improved Genetic Algorithm Approach to Energy-Efficient Flow Shop Scheduling With Job Due Dates, Smart And Sustainable Manufacturing Systems, 1 (1), 100 120
- Saurabh P, Nayak, A., Kumar A, Cheikhrouhou N. and Tiwari MK (2017) An integrated decision support system for berth and ship unloader allocation in bulk material handling port, Computers and Industrial Engineering, 106, 369 399
- Nayak, A., Rodrigo R., Lee, S and Nof S.Y. (2016) Resource sharing in cyber-physical systems: modelling framework and case studies, International Journal of productions Research, 54 (23), 6969 6983
- Mohapatra, P., Nayak, A., Kumar, S.K. and Tiwari, M.K. (2015) Multi-objective process planning and scheduling using controlled elitist non-dominated sorting genetic algorithm, International journal of production research 53 (6), 1712-1735
- Raj, R, Wang, J.W., Nayak, A., Tiwari, M.K., Han, B., Liu, C.L., Zhang, W.J. (2015) Measuring the resilience of supply chain systems using a survival model, IEEE Systems Journal 9 (2), 377-381.
- F.T.S. Chan, Nayak, A., Raj, R., Chong A.Y.L. and Tiwari, M.K. (2014) An innovative supply chain performance measurement system incorporating research and development (R&D) and marketing policy, Computers & Industrial Engineering 69, 64-70

ASHUTOSH NAYAK

N Salisbury Street, West Lafayette, IN, 47906 — 765 409 6516

CONFERENCE PUBLICATIONS

- Nayak, A., Lee, S. and Sutherland, J.W. (2018) Dynamic scheduling for minimizing electricity cost and carbon footprint by integrating Lyapunov optimization with future predictions (**Accepted: CIRP, LCE 2019**)
- Nayak A., Lee, S. (2018) Using Reinforcement Learning for Dynamic Load Scheduling in Smart Grid, Institute For Operations Research and Management Science, 2018, Phoenix, AZ
- Nayak A., Lee, S. (2017) Integrating Monte Carlo Simulation with Lyapunov Optimization for Dynamic Load Scheduling, Institute For Operations Research and Management Science, 2017, Houston, TX
- Nayak A., Lee, S. (2017) Real-Time Load Scheduling Using Lyapunov Optimization for Time Shift Able Loads With Time Window Constraints, Industrial and Systems Engineering Research Conference, Pittsburgh, PA
- Nayak, A., Kim, K. and Lee, S. (2016) Modeling Job Shop Scheduling for Peak Electricity Load Constraint, Industrial and Systems Engineering Research Conference, Anaheim, CA
- Pratap, S., Nayak, A., Cheikhrouhou, N., Tiwari, M.K. (2015) Decision support system for discrete robust berth allocation, IFAC-PapersOnLine 48 (3), 875-880

*: Under preparation