Implementation of Energy Hub Management System for Residential Sector

by

Hussin Hassen

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2010

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

This thesis is concerned with the implementation of a proposed Energy Hub Management System (EHMS) mathematical model for residential appliances under Time of Use (TOU) based electricity rate structure. The objective is to shift the residential electrical energy consumption during periods of high grid energy demand to low demand periods subject to operational constraints. The customer benefits from reduced daily energy consumption and consequent reduction to its cost with minimal effect on the comfort level.

Two scheduling periods are considered simultaneously. The first is a 24-interval schedule with one-hour time steps for appliances such as a dishwasher, clothes washer and dryer. The second is a 96-interval schedule with 15-minute time steps for other appliances such as refrigerator, freezer and water heater. Each appliance has been modeled as a discrete time linear dynamic system and the objective of this thesis is to make these models work in a real world situation by determining realistic estimations of the model parameters and constants. It is vital to properly calculate the mathematical model parameters as they have direct impact on the results. Minor modifications to some domestic appliance models were proposed to make the practical implementation easier. It was found that while some parameters in the mathematical model can be easily calculated based on thermodynamic equations, other parameters are hard to be calculated; therefore, a practical procedure was proposed to measure these parameters. An experiment on a small refrigerator was carried out to validate the refrigerator mathematical model and parameters measurement procedure.

The resulting model is a mixed integer linear problem (MILP) and was solved using GNU Linear Programming Kit (GLPK) freeware solver. The performance of GLPK was found to be satisfactory as compared to the commercial solver CPLEX, and was particularly suitable for practical and commercial implementations.

Acknowledgements

I would first like to thank God who gave me the health, strength and patience to complete this job. I would like also to express my sincere gratitude to my supervisors Professor Claudio Canizares and Professor Kankar Bhattacharya for their excellent guidance and complete support.

This project was carried out as a part of the Energy Hub Management System (EHMS) project at the University of Waterloo, funded by Ontario Center of Excellence Inc. (OCE), Hydro One, Milton Hydro and Energent. Funding received from the project to carry out this research is gratefully acknowledged.

I would like to thank Mohammad Chehreghani and other project team members for their cooperation and support. Special thanks to Chris Asselin from University of Waterloo for his proof reading of this work. Finally I would like to thank my wife for her patience, care and encouragement to finish this work.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AC : Air Conditioning
AMPL : A Modeling Language for Mathematical Programming
API : Application Programming Interface
COP : Coefficient of Performance
CPP : Critical Peak Pricing
CR : Charging device of PV battery
DLC : Direct Load Control
DR : Discharging device of PV battery
DRY : Dryer
DSM : Demand Side Management
DW : Dishwasher
EHMS : Energy Hub Management System
ESD : Energy Storage Device
FR : Fridge
GLPK : GNU Linear Programming Kit
GMPL : GNU Mathematical Programming Language
GNU : General Not Unix, a computer operating system
GPL : General Public License
H : Heating
HOEP : Hourly Ontario Electricity Price
HVAC : Heating, Ventilation and Air Conditioning
IL : Illumination Level
LDC : Local Distribution Company
LI : Lighting
LMS : Load Management System
LP : Linear Programming
MILP : Mixed Integer Linear Programming
MPS : Mathematical Programming System file format
PV : Photo Voltaic energy source
pump : Pool pump
RTP : Real Time Pricing
Stv : Stove
TOU : Time of Use
W : Washer
WH : Water Heater

Nomenclature

$S_i(t)$	Binary decision variable for i appliance ON/OFF status at time t
$U_i(t)$	Binary start up dummy variable for i appliance at time t
$D_i(t)$	Binary shutdown dummy variable for i appliance at time t
$\theta_i(t)$	Temperature of i appliance at time t
$SL_i(t)$	Storage level of i appliance at time t
$IL(t)$	Illumination level of the house at time t
$ESL(t)$	Energy storage level at time t
C_t	Cost of energy at time t
P_i	Rated power of i appliance
$AL(t)$	Activity level at time t
$AL_i(t)$	Activity level of i appliance at time t
$HWU(t)$	Average hourly hot water use at time t
MUT_i	Minimum up time of i appliance
MDT_i	Minimum down time of i appliance
$MSOT_i$	Maximum successive operation time of i appliance
EOT_i	Earliest operation time of i appliance
LOT_i	Late operation time of i appliance
ROT_i	Numbers of ON decisions of i appliance
MAT_{GAP}	Maximum time gap
θ_i^{up}	Upper limit of temperature of i appliance
θ_i^{low}	Lower limit of temperature of i appliance
Cap_i	Capacity of storage level of i appliance
$IL_{out}(t)$	Illumination of the house due to outdoor source (sun light) at time t
$IL_{req}(t)$	Required illumination at time t
ESL_{ESD}^{min}	Minimum energy storage level of the ESD
ESL_{ESD}^{max}	Maximum energy storage level of the ESD
$Discharg_{ESD}$	Discharged energy from the ESD during one time interval
$Charge_{ESD}(t)$	Charged energy into the ESD at time t
LPN	Large positive number
A	Set of appliances work on TA time schedule
B	Set of appliances work on TB time schedule
TA	Time schedule set {1: 96}
TB	Time schedule set {1: 24}
J	Objective function (total cost)
i	Index of appliances
t	Index of time
β_i	Beta parameter of i appliance

α_i	Alpha parameter of i appliance
γ_i	Gamma parameter of i appliance
T_i	Set of periods in which appliance i operates
$Ppv(t)$	Power generated by PV panel at time t
$P_{CHR}(t)$	Charging power fed into the batteries of PV system at time t
$BATESL^{min}$	Minimum PV battery storage level
$BATESL^{max}$	Maximum PV battery storage level
$BATESL(t)$	PV battery storage level at time t
C_P	Specific heat of water
H^*	Standby heat losses for water heaters
R	Water heater R-value
Q_{sen}	Sensible heat
Q_{lat}	Latent heat
h	Enthalpy
ρ	Density
m	Mass

Chapter 1

Introduction

1.1 Motivation

The Ontario government is phasing out coal fired generation plants to reduce green house gas emission. At the same time, nuclear generating facilities are reaching the end of their life [1], creating an urgent need to add new generating capacity or reduce consumption during peak periods or both. The long-term demand growth and generation retirements in Ontario are shown in Figure 1.1.

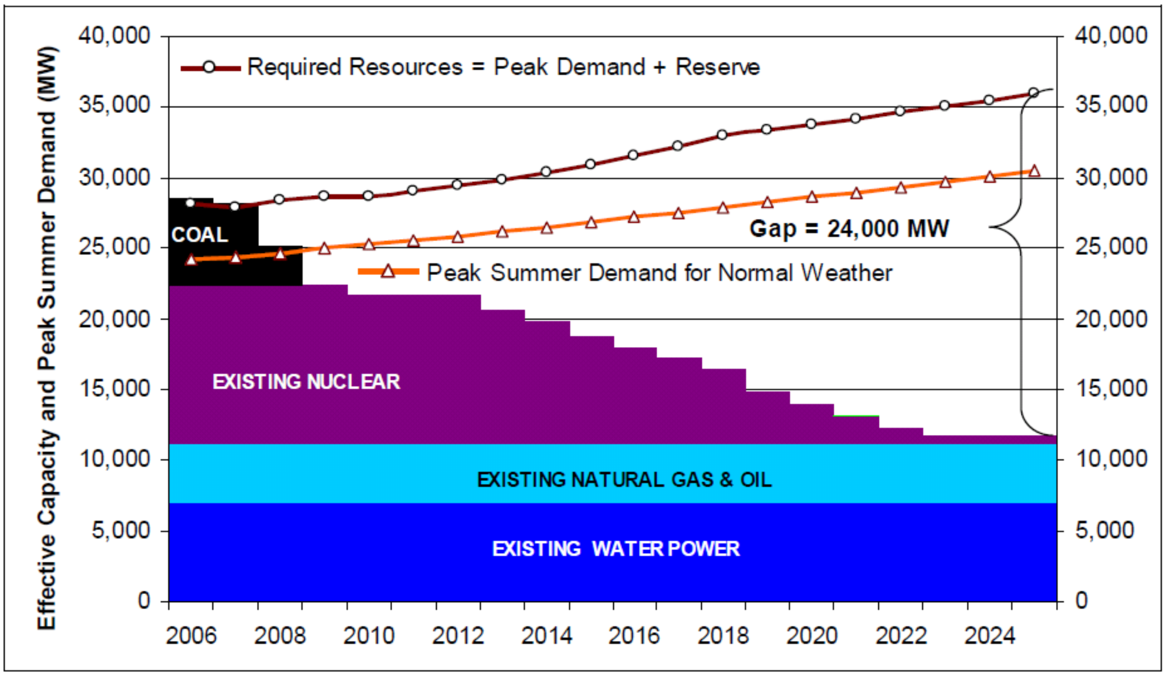
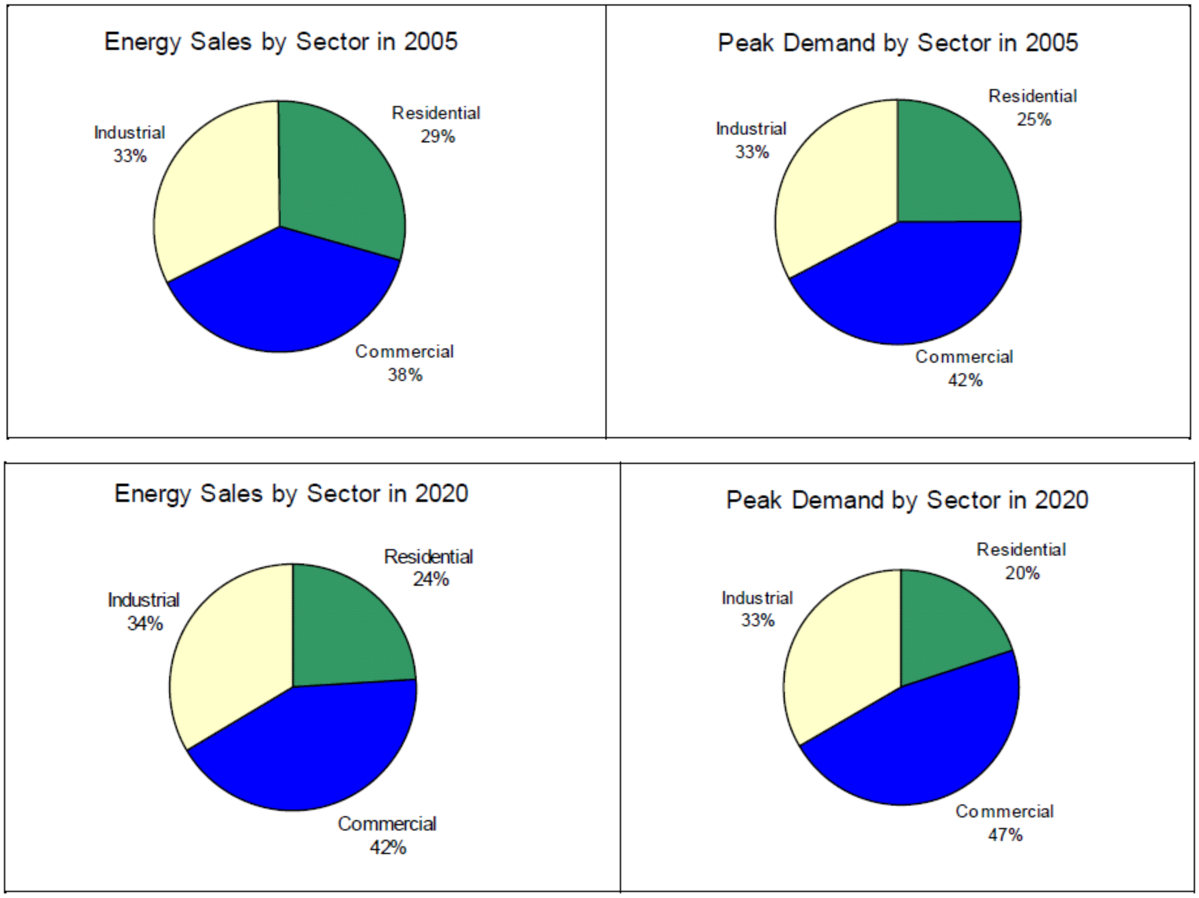


Figure 1.1. Demand growth and generation retirements in Ontario [1].

The residential sector took a 25% share of the peak electricity use in Ontario and consumed 29% of the electrical energy in 2005 [2], as shown in Figure 1.2. Although this share is expected to decline to a 20% of peak demand and 24% of energy use in 2020, it will still be accounting for a considerable amount of energy consumption. Therefore, controlling the residential end-use electricity demand can have a significant impact on reducing the peak demand.


Energy Sales by Sector in 2005

Peak Demand by Sector in 2005

Energy Sales by Sector in 2020

Peak Demand by Sector in 2020

Figure 1.2. Ontario electrical energy and peak demand by sector in 2005 and 2020 [2].

The break-up of residential electrical energy use in Ontario in 2007 is shown in Figure 1.3 [3], where it can be observed that residential heating, which is 30%, is the most significant contributor to electricity consumption in Ontario homes, followed by air-conditioning (space cooling) and lighting loads, which are 14% and 13% respectively. An important point to be noted here is that although dishwasher and cloth washer loads contribute only 0.5% each to the total electrical energy usage, these loads are of high power ratings that appear during a short period of time; therefore, these loads contribute to increase peak demand if not scheduled appropriately.

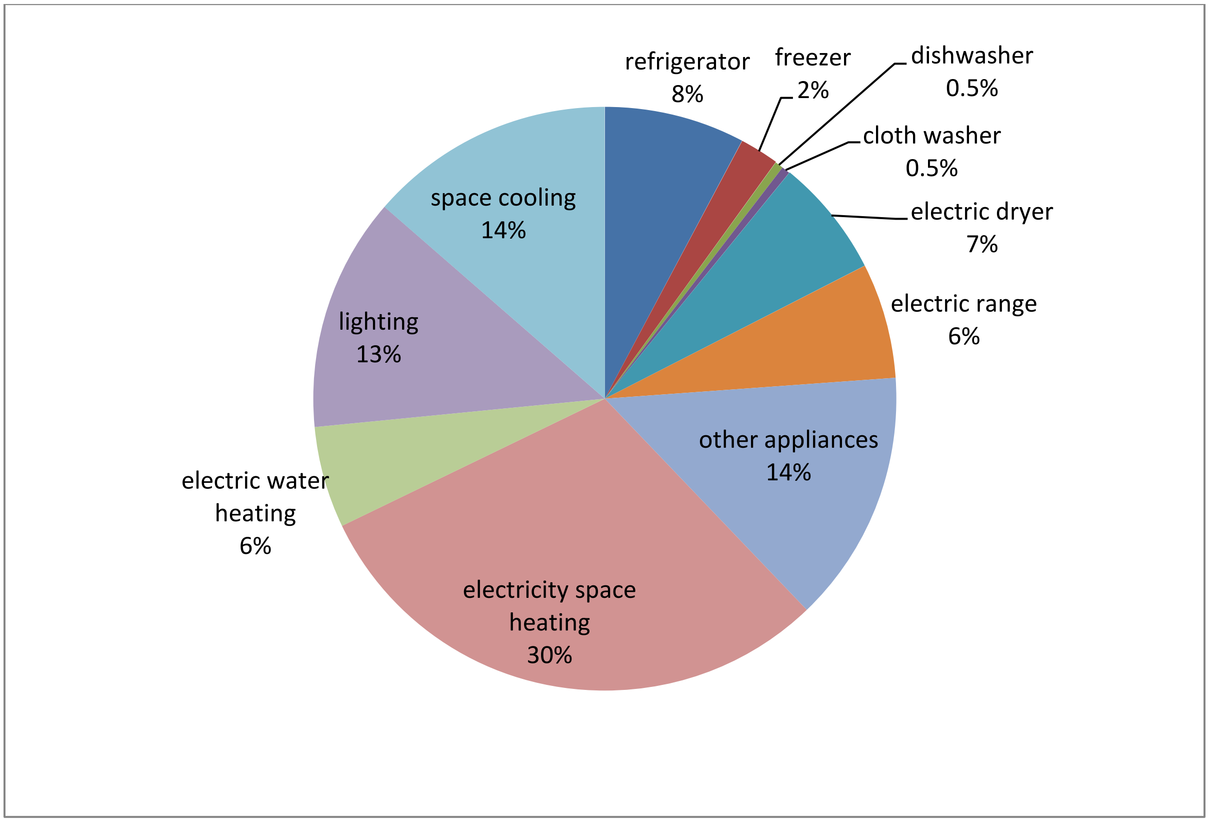


Figure 1.3. Percentage electrical energy use in residential sector in Ontario in 2007.

1.2 Demand Side Management

The demand for electrical energy has been rapidly increased due to economic developments and population growth. Moreover, electricity demand varies significantly over time due to various reasons such as extreme weather conditions. Electrical energy supply and demand should be balanced for consistent power delivery across the transmission grid. As demand comes close to generation capacity, the operating reserve capacity shrinks making the power system vulnerable to outages. More generating plants need to be built in order to circumvent such a situation. But this solution is restricted by economic and environmental constraints; therefore, it has been realized that modifying the usage of electrical energy would improve and flatten the load profile, increase the electrical grid reliability, reduce generation cost, defer the construction of new power plants or avoid operating standby emergency generators. Six load shaping objectives were stated in [4], which are: peak clipping, valley filling, load shifting, flexible load shaping, strategic conversion and strategic load growth, as shown in Figure 1.4. The first four load shaping objectives would be achieved using the proposed techniques in this thesis.

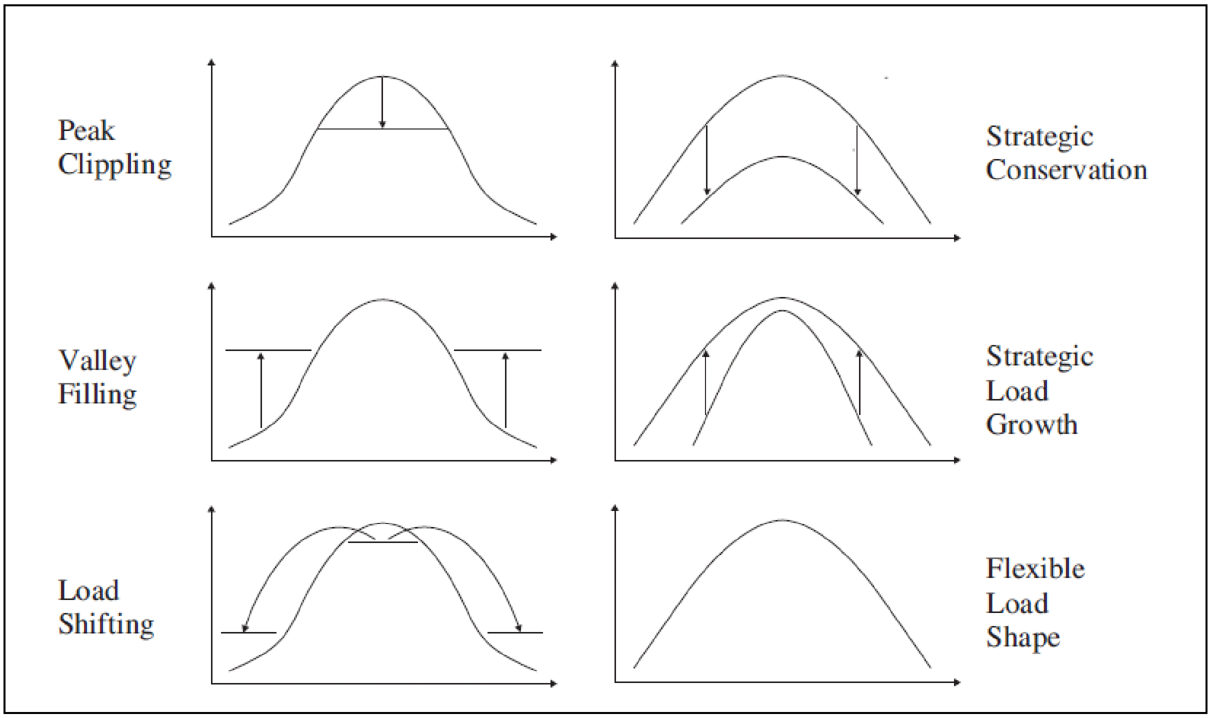


Figure 1.4. Load shape modifications objectives [4].

1.3 Smart Grids, Smart Meters and Smart Domestic Appliances

A Smart Grid is intelligent, efficient, accommodating, motivating, opportunistic, quality-focused, resilient and green [5]**.** The progress in internet technology, data communications, optimization methods and digital hardware/software engineering have made the development of the "smart grid dream" attainable.

Smart meters are new residential energy consumptions meters that are an important part of the smart grid. These meters can measure several parameters such as electricity consumption across different time periods. They can also display the electricity price and can communicate with the local distribution company (LDC) and the in house EHMS which can control the operation of individual domestic appliances. All separately metered residential properties in Ontario are required to be fitted with smart meters by 2010 [6].

Smart Domestic Appliances are those appliances that have the ability to reschedule their operation by themselves without the need for an external controller. Some of these appliances, for example, have frequency sensitive devices that can sense the deviation of the source voltage frequency on the local electric socket and switch themselves off for a few minutes when the frequency is low. Time of use (TOU) price can also encourage appliance manufacturers to make smart appliances that have, for example, smart thermostat units which enable the customer to enter the electricity price schedule in their region and program them to work at multiple set point schedules without the need for two way communications or an external controller. However, these smart appliances are typically not “smart enough” to communicate with each other to generate the overall optimum schedules as an energy hub management system (EHMS) can do.

1.4 Electricity Price Structures

To encourage the reduction of energy consumption during on-peak periods, a dynamic pricing scheme may be implemented by the utilities. There are different dynamic pricing structures such as time of use (TOU) and real time pricing (RTP). TOU pricing is going to be implemented in Ontario, and thus, the analysis in this thesis will be based on this pricing structure, though the mathematical model can be used for RTP scheme as well. In the context of Ontario, the TOU electricity price per kWh varies in three different time periods of the day namely, On-peak, Mid-peak and Low-peak periods as shown in Figure 1.5.

The RTP reflects the real cost of electricity in the wholesale market. The electricity price varies continuously based on the utility’s load, power market and power producers who participate in satisfying the demand [7]. RTP schemes are more efficient load management methods, yet their implementations require considerable technology investments [12]. An example of RTP over 24 hours is shown in Figure 1.6.

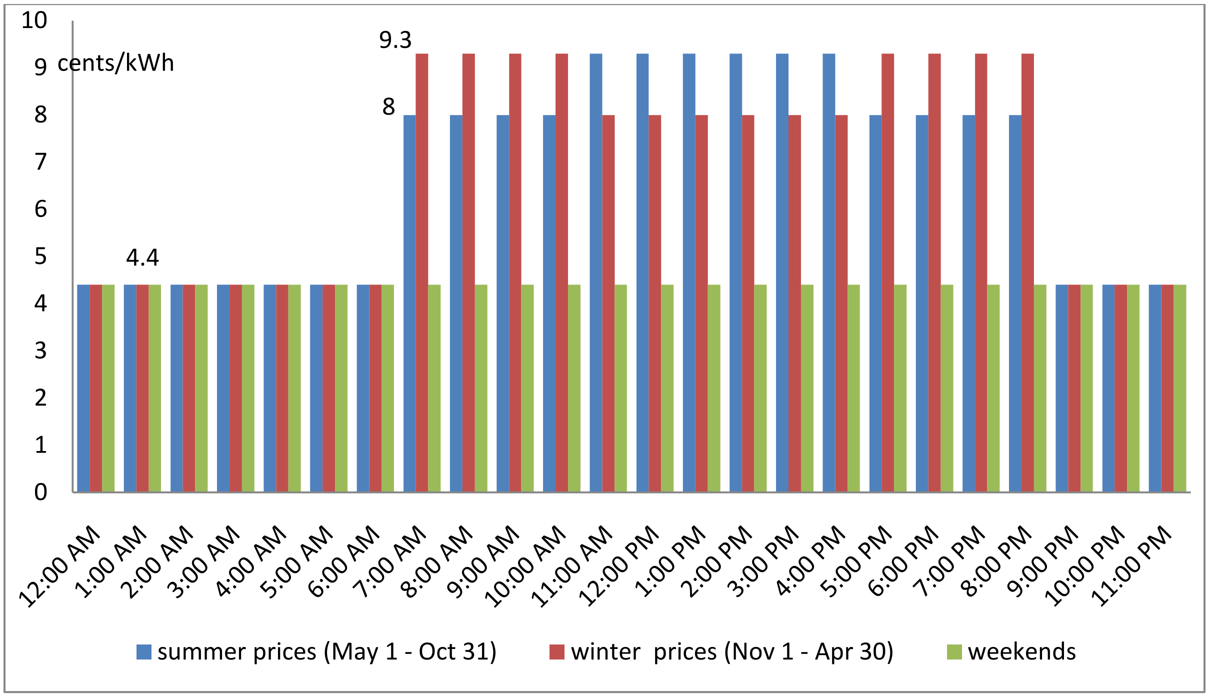


Figure 1.5. Ontario electricity TOU prices (effective Nov. 1, 2009).

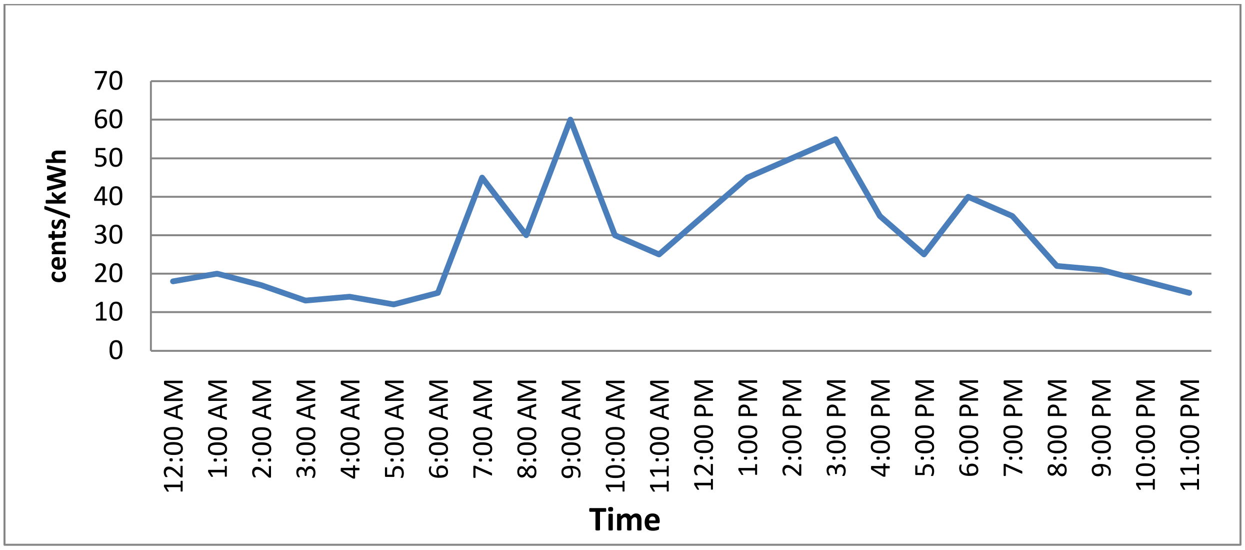


Figure 1.6. Example of RTP (real time pricing) scheme [7].

1.5 Energy Hub Management Systems

Figure 1.7 shows the residential energy hub management system proposed in [8]. Several household micro-hub systems are going to be connected to a main residential macro-hub system. The mathematical optimization model will empower each individual micro hub to smartly optimize energy consumption and reduce energy cost from the customer's point of view. On the other hand, the mathematical optimization model in the macro hub will optimize the energy consumption from both the utility's and customer's point of view [9]. This will result in a "multi-level" optimization problem. This thesis will concentrate on the practical implementation of the residential micro-hub system.

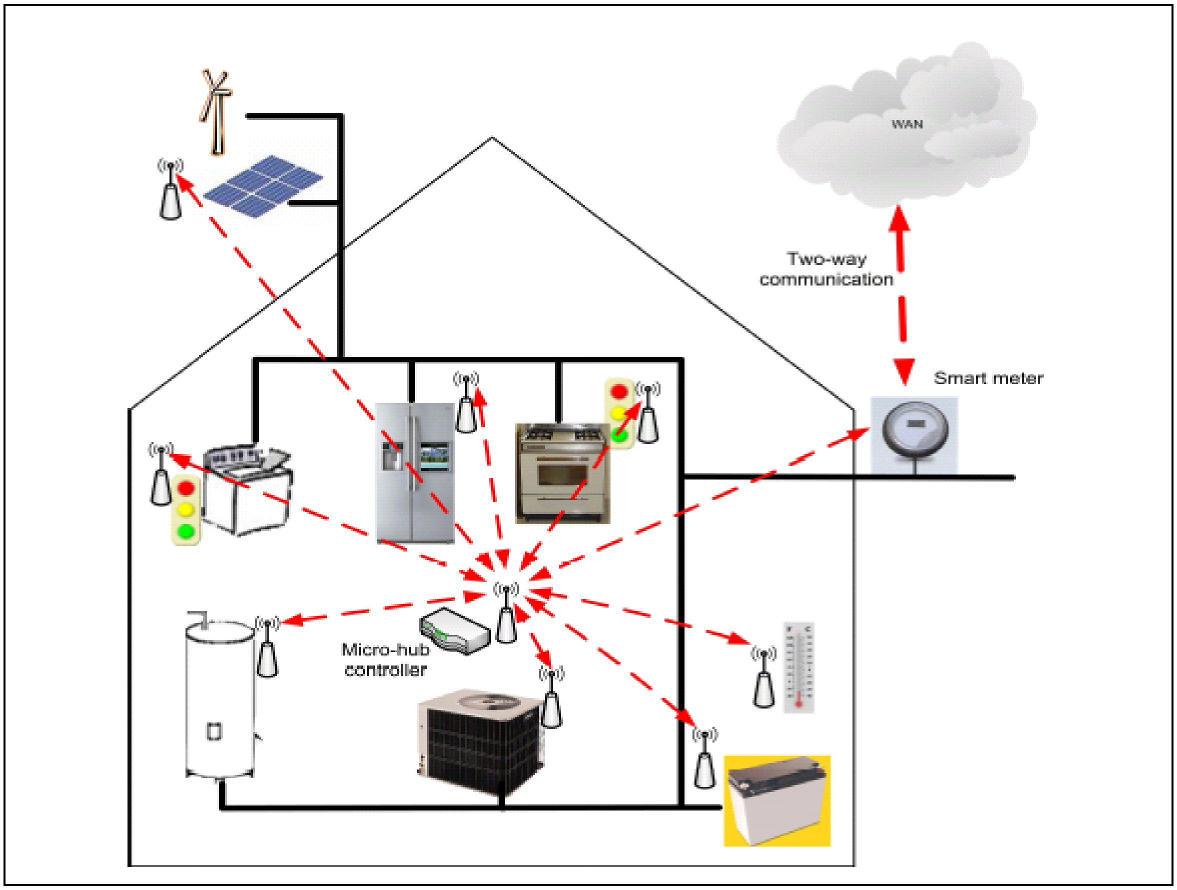


Figure 1.7. Residential energy hub system [9].

1.6 Linear Programming and Mixed Integer Linear Programming Problems

Mathematical optimization means to find the minimum, in case of minimization, or the maximum, in case of maximization, of an objective function by assigning a set of values to the variables of the function. For any specific objective function there may be many local and one global optimal solution. Linear programming (LP) problem with continuous variables can be mathematically stated in the following general form:

$$\min\ \mathrm{z} = c^T x$$

$$s.t.: Ax \leq b$$

$$x\ \geq 0$$

If some of the decision variables are integer variables, which is the case in mathematical model discussed in this thesis, the problem becomes a mixed integer linear programming (MILP) problem. Within each MILP problem, there is an LP problem, called LP relaxation, obtained by relaxing the integer constraint on the variables. MILP problems are difficult to solve in general; however, the past few years has seen an increase in the quantity and quality of software designed to solve MILP problems [10]. Non-commercial MILP software can't match the speed or strength of the commercial ones; however, they are more extensible, easier to customize and can provide a viable alternative to their commercial counterparts. Branch and bond method is the most common technique used to solve MILP problems by dividing the original problem to a series of smaller sub problems and then recursively solves each sub-problem. This is the technique used in this thesis to solve the MILP model considered here.

1.7 Major Implemented Load Control Projects

The authors in [11] and [12], conducted a comprehensive survey for the major direct load control projects implemented worldwide, which are summarized next:

- The Florida Power Corporation in the US implemented a full scale Direct Load Control (DLC) program in 1984 with over 52,000 residential customers and 500 commercial customers participating in the program. Water heaters, air conditioners, central heating systems, and pool pumps were cycled off during winter morning peak periods and summer afternoon peak periods. Control instructions were sent from the Load Management Office to appliances by a one-way communication system using VHF radio. Two DLC strategies were implemented: “normal” for expensive emergency units need to generate power and “scram” for severe conditions in the power system such as frequency fluctuations. Participants in the program received a financial incentive. Relatively high customer satisfaction was obtained.

- The Florida Power & Light Company (FPL) in the US, has implemented a direct load management system (LMS) with two-way power-line communication for more than 712,000 users. About 1000 MW of peak demand savings can be directly attributed to this project. In the system, water heaters and swimming pool pumps, central air conditioners and central strip heaters were controlled. Participants were paid yearly financial incentives for each appliance under control.
- Fifty customers participated in the demand limit control experiment in 1981 carried out by the Omaha Public Power District (OPPD). The residential demand controllers in each single customer location, used to control heat-storing loads and central air conditioners, central heaters, water heaters, and electric clothes dryers. Customers could negate the effects of the utility control signal by a control override button; however a penalty of $0.50 was applied for each use of the control override. A specific demand rate was applied for the participants that encouraged a high load factor through a demand charge. No other financial incentive was offered.
- The Korean Electric Power Research Institute (KEPRI) has performed a test of direct load control for air conditioners to reduce the summer peak load generated by air conditioners. One-way communication using VHF wireless network and an internal transceiver was used. The customers were given an incentive of 10-20% price reduction when they purchased controllable air conditioners. In the field test, room temperature rose by up to 1.2°C. A survey showed that 69% of customers were satisfied with the new type of air conditioners.
- The Kyushu Electric Power Company in Japan implemented an experiment from 1994 to 1999 to reduce the peak summer load by controlling air conditioning loads using a radio signal. Customers were divided into direct, indirect or no-control groups of 400 customers each group. In the direct control group, the customers were controlled with ON/OFF mode or constant temperature mode. In the indirect control group, the customers were given electricity tariff and the system demand curves in real-time and the customers were expected to reduce power consumption at high tariff periods. In the no control customer group, the demand curve was monitored in order to compare with other groups and to evaluate the effectiveness of the

load controls. The results of the test indicated a peak demand reduction of between 50 W and 100 W per customer on days with a maximum temperature in excess of 30°C, for both the direct control and indirect control customer groups.

- The University of Manchester explored the use of a quasi real-time two-way communication infrastructure for domestic appliance load shifting and reduction. The system operator determined a desired load curve taking into consideration the capacity of the distribution network, and then calculated the optimal control actions before the start of the control period. When a customer pressed the ON button on an appliance, the communication system sent the connection request signal to the system operator. The system operator checked the list of the control actions and sent either the direct connection permission or the control signal (shift or curtailment) to the particular appliance.
- Load management to minimize frequency fluctuations has been carried out in some small isolated power systems on various islands in Europe such as Rum Island with Hydro/Diesel system. The power system was facing blackouts due to temporary load peaks; therefore, a load shedding controller was used. The controller measured system frequency, compared it with a setting threshold and sent a signal to disconnect some appliances when the frequency was lower than the threshold. About 42 appliances for 15 houses were assigned priorities using a total of 11 frequency thresholds. The devices with the lowest priority were given the highest frequency threshold, so they would be the first to be shed in the event of an overload threshold.
- The Electricite De France (EDF) operated a successful program of TOU pricing in which 30 million customers participated. In addition to TOU, EDF introduced a critical peak pricing (CPP) program called “Tempo” in which the year was divided into three types of days: Tempo Blue, Tempo White, and Tempo Red. The electricity prices in Tempo Blue were cheaper than the normal TOU prices, while in Tempo Red they were the most expensive; the prices in Tempo White were slightly higher than the normal TOU. About 300 days a year were Tempo Blue, 43 were Tempo White, and 22 days only were Tempo Red.
- The Gulf Power Company implemented an experimental program in Florida with TOU pricing which was able to reduce the peak demand during critical periods by 24%. Within the program, customers were provided with smart thermostats that automatically adjust the

temperature depending on a price signal. About 96% of customers where fully satisfied with this program.

- An experiment was carried out by the state of California to test customer response to a variety of pricing options, including TOU and CPP rates. It was found that CPP rates were very effective in reducing customers' peak demands, as customers were likely to reduce air conditioning usage during on-peak periods and shift laundry, dishwashing and cooking activities from on-peak periods to some off-peak periods.
- The Chicago Community Energy Cooperative and the local electricity utility implemented a three-year RTP pilot project for residential customers in 2003. The project was designed to estimate the extent of customer response to hourly energy pricing. It was reported that 25% of aggregated demand reduction was achieved during the notification period, and 20% of customers' monthly bill was saved. Customer satisfaction was very high in this program.
- After a two-year pilot project started in 1992, Georgia Power operated a successful RTP program that was able to reduce the demand by 17%, which is approximately 800MW, during emergency conditions. The program mainly targeted commercial and industrial customers. The configuration of the RTP rate consisted of two parts: baseline rate and the rate difference between the baseline rate and the hourly real time pricing rate.

It can be concluded from the above survey that residential load management programs have been widely and successfully implemented throughout. However, more improvements are required in this field, like the one proposed in this thesis, by integrating all different household appliances of different behaviors into a unified optimization model with multiple objective functions. The progress in internet technologies, data communications, smart meters and smart grid will help to accomplish this goal.

1.8 Objectives

In this thesis the objective is to develop the proper framework such that the mathematical model, proposed in [9], is workable and can be implemented in a simple hardware device which can be installed in a household for such appliance control decision making. Therefore, the following are the objectives of this thesis:

- Develop proper realistic estimate of the model parameters for each residential sector appliance and parameters for the household.
- Identify an appropriate freeware solver to solve the MILP model with the same level of computational efficiency as that of the commercially available CPLEX solver.
- Test the performance of the identified freeware solver for its suitability in practical applications.
- Undertake a practical case study for one particular house-hold appliance to validate the proposed model and control strategy.

1.9 Thesis Organization

This thesis is divided into five chapters. In Chapter 2, the implementation details and parameter estimation of the mathematical model of each individual appliance is discussed. In Chapter 3, the selection of the chosen solver for practical implementation is discussed and simulation and experimental results are presented. Finally; in Chapter 4, a summary of the thesis and main conclusions and contributions of this work are presented; some possible future work is also discussed in this chapter.

Chapter 2

Residential Appliance Models and Parameters

In this chapter, the residential sector model is presented taking into account the operational constraints of each appliance. The concept of residential customer activity level and its estimation is also discussed. A detailed discussion on the procedure to estimate the required model parameters, which is one of the main contributions of this thesis, is presented here.

2.1 Objective Function of the Mathematical Model

The objective function of the residential micro-hub is to minimize the customer's daily energy cost, which is given by the following equation [9]:

$$\begin{aligned} min\ J = & \sum_{t \in TA} \left[\sum_{i \in A, i \neq LI, i \neq ESD} C_t(t) P_i S_i(t) + \sum_{i=LI} C_t(t) P_i IL(t) - \sum_{i=ESD} C_t(t) P_i S_i(t) \right] \\ & + \sum_{t \in TB} \left[\sum_{i \in B} C_t(t) P_i S_i(t) \right] \end{aligned} \tag{2.1}$$

The objective is to minimize J, i.e., the total energy cost of set A appliances working over scheduling horizon TA (96 fifteen-minute intervals), and set B appliances working over scheduling horizon TB (24 one-hour intervals), C_t is the hourly energy cost, P_i is the power consumption of each appliance, $S_i(t)$ is the ON/OFF status of each appliance and t and i represent the time and appliance index respectively. Since the energy storage devices, ESD, represent any sustainable energy source (wind or solar), which can provide free energy as will be explained later, their contribution is negative in the cost function; in other words, the optimization model will try to maximize the use of power generated from the ESD, and at the same time, to shift the energy consumption of all other appliances away from on-peak periods as much as possible subject to operational constraints. It is important to note in the objective function equation that all appliance's operating decision variables, $S_i(t)$, are binary variables except that for lighting LI where the decision variable, $IL(t)$, is an integer that represents the number of zones illuminated by electricity. More details on the mathematical model and constraint equations of lighting loads will be discussed in section 2.11.

2.2 Activity Level

In order to include the effect of household occupancy and activity on appliances energy consumption pattern, a new time dependent index called Activity Level, $AL(t)$, is proposed in [9]. Since house occupancy and activity has different effects on different electrical appliances, an appropriate coefficient β_i is introduced for each appliance to reflect the weight of this effect. The activity level index was used in the mathematical models of the refrigerator, space heating/cooling appliances, dishwasher, cloth washer and dryer only. The challenge is then to calculate the activity level index and its associated coefficient β_i. It should be noted that the household occupancy and activity is also affected by the season of the year and day of the week. Therefore; it was proposed in [9] to extract the activity level information from hourly household energy consumption data measured on a similar day of a previous year or to use statistical methods similar to that used in electrical load forecasting. The measured or forecasted data can then be normalized with respect to the total energy consumption of the day, which is assumed to be 100%.

More accurate procedures are needed to measure or calculate the activity level index or its associated β_i coefficients in the EHMS implementation, because it has a direct influence on the optimum decisions. Hourly dwelling energy consumption of a particular day of the year does not necessarily match energy consumption of a similar day of next year. Moreover, appliance usage activity might not be easily extracted from normalized hourly energy consumption by simply multiplying it by a constant coefficient such as β_i. In that case, a time varying coefficient or other methods are needed to extract the appliances' time of use activities as shown here for refrigerators, washers and dishwashers.

Smart meters that are going to be supplied to each house participating in the EHMS project can provide a wealth of data, including energy consumed each hour or even in each fifteen minute interval. Therefore, the measured data of the previous week can be used to predict the energy consumption on a similar day of the present week. This can lead to a more reliable forecasting than using data of the previous year or data obtained from a sophisticated stochastic calculation which requires large amounts of parameters including customer's income, number and ages of occupants, environment and other demographic data. Hourly energy consumption forecasting based on the previous week measurement can be further improved by including the effect of few other variables, such as the temperature forecast of the current day and fuel type for the space heating/cooling system used in the house. The mathematical equations for calculating the activity level is given by:

$$AL(t) = EEC(t) \div \sum_{t=1}^{t=T} EEC(t) \tag{2.2}$$

where $EEC(t)$ is the household electrical energy consumption of the previous week. Figure 2.1 shows an example of hourly household electrical energy consumption in a Canadian house [13], where one can see peak consumption in the afternoon and evening hours. To extract the activity level, the energy consumption data were normalized to the total consumption of that day according to (2.2) and the resulting data are presented in Figure 2.2.

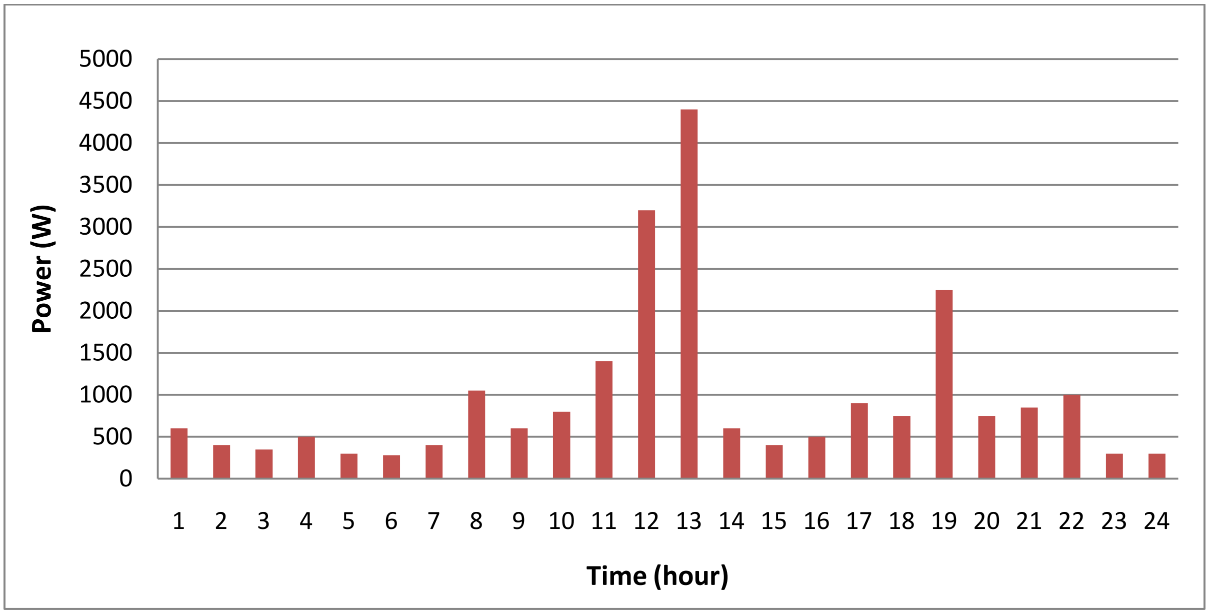


Figure 2.1. Example of hourly household electrical energy consumption [13].

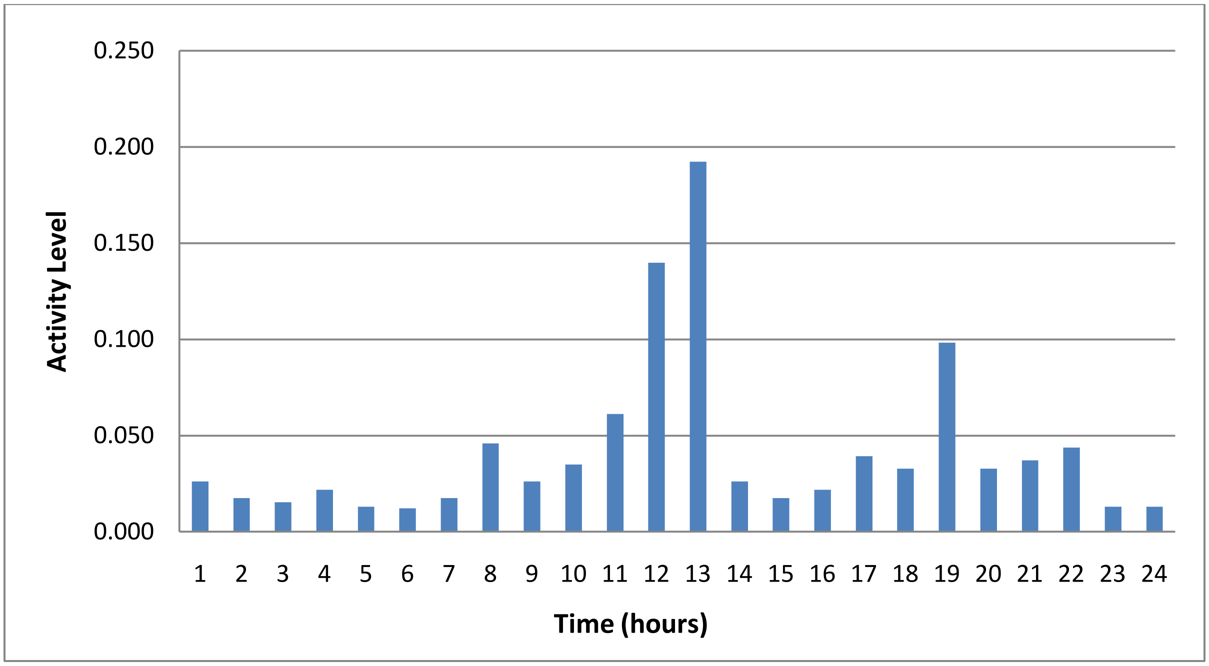


Figure 2.2. Normalized household electric energy consumption (activity level).

2.3 Water Heater

Domestic water heating is the second largest energy end-use for Canadian households after space heating, accounting for approximately 19% of total household energy consumption in 2007 [3]. The total residential energy used in Ontario in 2007 for water heating was 105 PJ, out of which only 8.1% was from electricity and remaining from other sources. Electrical energy consumption from residential electric water heaters only was approximately 6% of the total electrical energy used (see Figure 1.3).

Storage tank water heaters are the most common type used in Canada. These systems heat and store water in a tank so that hot water is available at any time. The thermal storage capacities of water heaters make them perfect target for load management programs. In this thesis only electric storage tank water heaters will be considered.

Most electric water heaters have two elements and two thermostats. When hot water is drawn from the top of the tank, cold water enters the bottom of the tank activating the upper thermostat and the top portion of the water will heat up to the setting of the thermostat [14]. Once the temperature reaches the setting of the upper thermostat, the thermostat will then flip down the electric power to the lower heating element casing the bottom portion of the tank to heats up until the water reaches the setting on its thermostat set-point. When hot water is being used, cold water enters the bottom of the heater, and the bottom element will begin to heat the cold water. If lots of hot water has been used, the upper thermostat will take priority and the top portion of the heater will be heated to the set point, then power will be flipped to lower heating element to heat the lower portion. Thermostats have a safety switch, usually a red button, that switches off the electricity once the water temperature goes too high (above 82^{o}C); the red button must be reset before the water heater elements will work. The upper and lower thermostats are usually factory set at 60^{o}C in Canada.

2.3.1 Operational Constraints

The mathematical model of the water heater in [9] corresponds to the following set of equations:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = WH, \\ 0 & \text{if } t \notin T_i\,, i = WH, \end{cases} \qquad EOT_i \;\le\; T_i \;\le\; LOT_i \tag{2.3}$$

$$S_i(t=1) = \begin{cases} 1 & \text{if } \theta_i(t=0) < \theta_i^{low} \\ 0 & \text{if } \theta_i(t=0) > \theta_i^{up} \end{cases} \qquad \forall t \in T_i\,, i = WH \tag{2.4}$$

$$\theta_i^{low} \le \theta_i(t) \le \theta_i^{up} \qquad \forall t \in T_i\,, i = WH \tag{2.5}$$

$$\theta_i(t) = \theta_i(t-1) - \beta_i\, HWU(t) + \alpha_i\, S_i(t) - i\left(1 - S_i(t)\right) \qquad \forall t \in T_i\,, i = WH \tag{2.6}$$

$$U_i(t) - D_i(t) = S_i(t) - S_i(t-1) \qquad \forall t \in T_i\,, i = WH \tag{2.7}$$

$$U_i(t) - D_i(t) \le 1 \qquad \forall t \in T_i\,, i = WH \tag{2.8}$$

$$\sum_{k=t}^{t+MUT_i} S_i(k) \ge MUT_i - LPN\left(1 - U_i(t)\right) \qquad \forall t \in T_i\,, i = WH \tag{2.9}$$

$$\sum_{k=t}^{t+MDT_i-\mathbf{1}} S_i(k) \le LPN\left(1 - D_i(t)\right) \qquad \forall t \in T_i\,, i = WH \tag{2.10}$$

where β_{WH} is the Effect of hot water usage on water temperature (^{o}C per liters of hot water usage), α_{WH} is the warming effect of one ON state of WH (^{o}C per time interval) and γ_{WH} is the cooling effect of one OFF state of WH (^{o}C per time interval). Equation (2.3) enables the customer to specify the period T_i, of the water heater operation between EOT_i (early operating time) and LOT_i (late operating time). Equation (2.4) ensures that if the water heater temperature, at $t = 0$, is less than the lower limit, as specified by the customer, the water heater state is ON in the first time interval and vice versa. Equation (2.5) ensures that water heater temperature is within the customer's preferred range. Equation (2.6) relates the temperature of water heater at time t to the temperature at the previous time interval $t-1$, hot water use $HWU(t)$ at time t, and the ON/OFF state of the water heating element. Equations (2.7) to (2.10) ensure that the water heater ON/OFF status will have a minimum predetermined successive numbers.

It was found that it is easier to implement (2.6) if it is modified as follows:

$$\theta_i(t) = \theta_i(t-1) - \beta_i\, HWU(t) + \alpha_i\, S_i(t) - \gamma_i \qquad \forall t \in T_i, i = WH \qquad (2.11)$$

This gives a practical and measurable meaning for the parameter γ_{WH}, which can be redefined as the effect of hot water heat losses on water temperature (^{o}C per time interval). Moreover, as the time interval has been selected to be 15 minutes, and since there is no problem for the heating element to change its state every 15 minutes interval (if necessary), one does not need to worry about the minimum numbers of successive ON/OFF times of the water heater. For this reason (2.7) to (2.10) can be ignored; therefore, the final mathematical model of the water heater to be implemented is given by the following set of equations:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i, i = WH, \\ 0 & \text{if } t \notin T_i, i = WH, \end{cases} \qquad EOT_i \le T_i \le LOT_i \qquad (2.12)$$

$$S_i(t=1) = \begin{cases} 1 & \text{if } \theta_i(t=0) < \theta_i^{low} \\ 0 & \text{if } \theta_i(t=0) > \theta_i^{up} \end{cases} \qquad \forall t \in T_i, i = WH \qquad (2.13)$$

$$\theta_i^{low} \le \theta_i(t) \le \theta_i^{up} \qquad \forall t \in T_i, i = WH \qquad (2.14)$$

$$\theta_i(t) = \theta_i(t-1) - \beta_i\, HWU(t) + \alpha_i\, S_i(t) - \gamma_i \qquad \forall t \in T_i, i = WH \qquad (2.15)$$

2.3.2 Calculation of Parameters

The parameter α_{WH} represents the rate at which water temperature increases when the heating element is in ON state for one time interval of 15 minutes. It can be calculated using thermodynamic equations as follows:

$$H_i = m\, C_p\, \alpha_{WH} \qquad (2.16)$$

where H_i is the amount of heat (in Joules) injected into the water when the heater is ON during one time interval, C_p is the specific heat of water = 4185 J/kg.^{o}C, α_{WH} is the amount of water temperature rise (in ^{o}C) per time interval, and m is the mass of the water inside the water heater tank in kg. (each one liter is approximately 1 kg). The amount of heat injected to the water when the heating element is ON for Δt seconds can be also calculated using the following equation:

$$H_i = P\ \Delta t \tag{2.17}$$

where P is the heating element power rating in watts and Δt is the time interval $= 15 \times 60 = 900$ seconds. From the above it follows that:

$$\alpha_{WH} = \frac{P\ \Delta t}{m\ C_p} \tag{2.18}$$

For example, if the heating element power rating is 3800 W and the water heater capacity is $184l$, then a temperature rise of $\alpha_{WH} = \frac{3800 \times 900}{184 \times 4185} = 4.44^{o}$C will be observed after an interval of 15 minutes, i.e. 900 sec.

The parameter γ_{WH} represents the temperature drop (in ^{o}C) due to heat losses through tank walls during Δt seconds. The lost (or rejected) heat (in joules) from a water tank is a function of hot water temperature, ambient temperature surrounding the water tank and the water tank shield resistance, and is given by:

$$H_r = m\ C_p\ \gamma_{WH} \tag{2.19}$$

Canadian's water heater manufacturers normally specify a parameter called the tank standby heat losses. This parameter, which is denoted by H_r^*, represents the amount of water tank heat loss to the ambient in joules per seconds. Therefore, the heat losses in joules after Δt seconds can be calculated as:

$$H_r = H_r^*\ \Delta t \tag{2.20}$$

This yields:

$$\gamma_{WH} = \frac{H_r^*\ \Delta t}{m\ C_p} \tag{2.21}$$

The value of H_r^* in (2.21) assumes that the water heater set point is 60^{o}C, and the water heater is inside a house with a surrounding air temperature of 20^{o}C. If this is not the case, then (2.21) should be modified as follows:

$$\gamma_{WH} = \frac{H_r^* \, \Delta t \, (\theta_{WH}^{set} - \theta_{amb})}{40 \, m \, C_p} \tag{2.22}$$

Water heater manufacturers in the US usually supply a parameter called the water heater R-value which represents the thermal resistance of the water heater shield in ft^2.hr.F^o/Btu. In this case, the British Thermal Units need to be used to calculate the heat loss rate in Btu/hr and then convert it to the SI units in J/s. This yields:

$$H_r^* = \frac{(\theta_{WH}^{set} - \theta_{amb}) \, A}{3.142 \, R} \tag{2.23}$$

where H_r^* is the heat loss rate in W or J/s, R is the water heater R-value (normally a number between 16 and 24), A is the water heater surface area in sq-ft., θ_{WH}^{set} is the water heater set point temperature in ^{o}F, θ_{amb} is the water heater surrounding ambient temperature in ^{o}F.

For example, if a standby loss of 71 W was specified by a Canadian manufacturer for a 184 l water heater, then (2.21) yield a drop in temperature of $\gamma_{WH} = \frac{71 \times 900}{184 \times 4185} = 0.083$^{o}C over a 15 minutes time interval. However, for a 208 l capacity water heater which has a 33.2 ft^2 surface area and a set temperature of 60^{o}C (140^{o}F) and an ambient temperature of 20^{o}C (68^{o}F) with R-value of 20, the γ_{WH} can be calculated using (2.23) and (2.21), for Δt = 15 min. = 900 s, as follows:

$$H_r^* = \frac{(140-68) \times 33.2}{3.142 \times 20} = 38 \text{ W}$$

$$\gamma_{WH} = \frac{38 \times 15 \times 60}{208 \times 4185} = 0.04$$

The β_{WH} parameter represents the temperature drop due to hot water consumption. If one liter of hot water is consumed during an hour, then $0.25\,l$ of hot water will be consumed during a time interval of 15 minutes (assuming an even hot water consumption rate within each hour). When 0.25 l of hot water is consumed, the same amount of cold water enters the water tank and mix with the remaining m kg (or m liters as each liter of water has a mass of 1 kg) of hot water in the tank, and the new water temperature θ_{WH}^{new} will be less than the original hot water temperature by β_{WH} amount, which is equal to:

$$\beta_{WH} = (\theta_{WH}^{set} - \theta_{WH}^{new}) = \frac{0.25 \left(\theta_{WH}^{set} - \theta_{WH}^{inlet}\right)}{m} \tag{2.24}$$

where m is the total mass of hot water inside the tank in kg, θ_{WH}^{set} is the water heater temperature set point and θ_{WH}^{inlet} is the temperature of the inlet cold water.

For a 184 l capacity hot water tank with a temperature set point of 60° C and an inlet cold water temperature of 10°C, the temperature drop is $\beta_{WH} = \frac{0.25 \times (60-10)}{184} =$ 0.068° C for a 15 minute time interval, per liter of hourly hot water consumption. Note that the inlet cold water temperature is variable depending on the time of the year and the location in Ontario.

In the above discussions it was assumed that the consumed hot water has a constant temperature of 60°C. This is actually a reasonable approximation as the used water temperature could be any value within the upper and lower limits.

2.3.3 Calculation of hourly hot water use

It is extremely difficult to measure or predict the hourly hot water use $HWU(t)$ in a house. For this reason, the formula proposed by Lutz et al in [15] is used here, which is an extended version of the original formula proposed by EPRI [16], and is given below:

$$\begin{aligned} HWU(t) = [\, & a_o + a_1\,(per) + a_2\,(age1) + a_3\,(age2) + a_4\,(age3) + a_5\,(therm) \\ & + a_6\,(tanksz) + a_7\,(wtmp) + a_8\,(atmp) + a_9\,(athome) \\ & + a_{10}\,(spring) + a_{11}\,(summer) + a_{12}\,(fall) + a_{13}\,(winter) \\ & - no_{CW} - no_{DW}\,]\,(1 - 0.621\,senior)\,(1 + 0.3625\,no_{pay}) \end{aligned} \tag{2.25}$$

where:

$HWU(t)$ is the hourly hot water consumption in l/hr

per is the number of persons in the household

$age1$ is the number of preschool children (0-5 yrs)

$age2$ is the number of school age children (6-13 yrs)

$age3$ is number of adults (14 yrs and over)

$therm$ is the water heater thermostat setting in°C

$tanksz$ is the water heater nominal tank size in l

$wtmp$ is the water heater inlet water temperature in °C

$atmp$ is the outdoor air temperature in °C

$athome$ is a dummy variable for the presence of adults at home during day (unemployed person)

$spring$ is a dummy variable for spring (1 if “spring”, zero otherwise)

$summer$ is a dummy variable for summer (1 if “summer”, zero otherwise)

$fall$ is a dummy variable for fall (1 if “fall”, zero otherwise)

$winter$ is a dummy variable for winter (1 if “winter”, zero otherwise)

no_{DW} is a dummy function indicates hot water saving when not owning a dish washer

$$no_{DW} = \begin{cases} (0.655\,per + 1.2635\ \sqrt{per}), & \text{if no dishwasher(for hrs } 5-9 \text{ pm)} \\ 0 & \text{otherwise} \end{cases}$$

no_{CW} is a dummy function indicates hot water saving when not owning a clothes washer

$$no_{CW} = \begin{cases} \left(2.2121\,per + 9.035\,\sqrt{per}\right), & \text{if no clothwasher(hrs } 7-9 \text{ pm weekdays)} \\ \left(1.4750\ \ per + 6.023\,\sqrt{per}\right), & \text{if no clothwasher(hrs } 8-11 \text{ am weekends)} \\ 0 & \text{otherwise} \end{cases}$$

$senior$ = 1 if this is a senior-only household, = 0 otherwise

no_{pay} = 1 if household does not pay for hot water, = 0 otherwise

a_1 to a_{13} are different constants for weekdays and weekends and are given in Table 2.1.

In equation (2.25), if hot water savings due to absence of dishwasher and washer is more than the consumption during some hours in the summer time, the net hot water consumption values during these hours becomes negative, which is not possible. Therefore, the dummy functions no_{DW} and no_{CW} of hot water savings when dishwasher and washer are not available were modified so that they are linearly dependent on the ambient temperature by assuming the savings are 100% when the ambient air temperature is -40^{o}C and 0% when it is +40^{o}C, which is a reasonable assumption. For the same reason, the hot water savings for clothes washer absence was also distributed over a wider period of time, which is 5-9 pm instead of 7-9 pm. The new no_{DW} and no_{DW} functions are:

$$no_{DW} = \begin{cases} (0.5 - 0.0125\,atmp)\ (0.655\,per + 1.2635\ \sqrt{per}), & \text{if no DW (for hrs } 5-9 \text{ pm)} \\ 0 & \text{otherwise} \end{cases}$$

$$no_{CW} = \begin{cases} (0.5 - 0.25\,atmp)\left(1.106\,per + 4.5175\sqrt{per}\right), \text{if no CW(hrs } 5 - \text{9pm) weekdays} \\ (0.5 - 0.25\,atmp)\left(1.475\,per + 6.023\sqrt{per}\right), \text{if no CW(hrs } 8 - \text{11am) weekends} \\ 0 & \text{otherwise} \end{cases}$$

An example of household hourly hot water consumption using (2.25) is provided in Figure 2.3 for a single family consisting of two adults, and two kids (4 & 6 years old), with one adult staying at home in winter when the outside temperature is -10^{o}C, and inlet water temperature at 8^{o}C, with water tank size of 185 l and no dishwasher available at home.

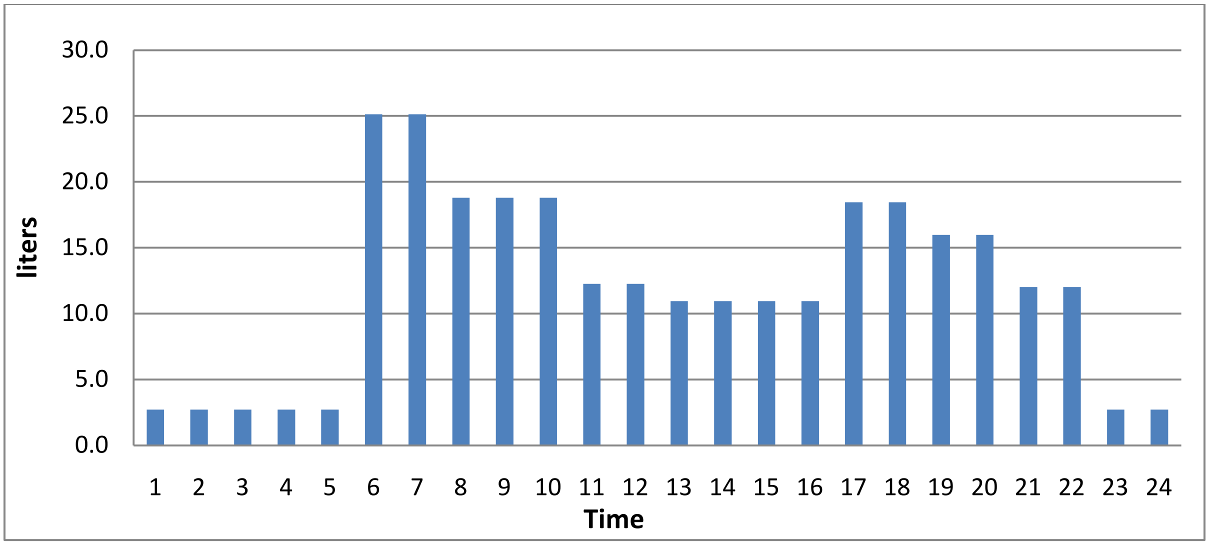


Figure 2.3. Example of hourly hot water consumption according to equation (2.25).

Table 2.1. Modified constants in EPRI hourly hot water use model [15].

WEEKDAY (SI units)

EPRI														EXTENDED			
hour	a0	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	a11	a12	a13	no_dw	no_cw	senior	no_pay
	(const)	(per)	(age1)	(age2)	(age3)	(therm)	(tank)	(wtmp)	(atmp)	(athome)	(spring)	(summer)	(fall)	(winter)	[1]	[2]	[3]	[4]
11--6	0	0.6163	0		0	0	-0.0017	0	0	0	0	0	0	0.5523	0	0	0.379	1.3625
6--8	2.0956	0	0	3.483	7.9861	0	0.0269	-0.5424	0.6603	-3.6609	0	-13.601	0	0	0	0	0.379	1.3625
8--11	0	0	1.0853	1.5331	2.4972	0	0	0	0	9.0418	0	0	-1.6353	2.1403	0	0	0.379	1.3625
11--1	-0.3876	0	0.9668	1.0849	2.0956	-0.0218	0	0	0	6.1986	0	0	-1.6834	1.5187	0	0	0.379	1.3625
1--5	-0.2907	0	1.979	1.2	2.3072	-0.0906	0.0083	0	0.0743	4.0228	0	0	0	2.5854	0	0	0.379	1.3625
5--7	0.7753	0	1.5679	2.0415	3.6018	0	0	-0.3134	0.357	5.3492	0	-3.6855	0	3.656	5.147	0	0.379	1.3625
7--9	4.4577	0	2.7456	4.4092	3.3455	-0.1015	0.0187	0	0.3523	0	0	-8.0527	-3.2509	0	5.147	26.9184	0.379	1.3625
9--11	4.2881	0	1.4434	3.4394	2.5135	-0.0436	0	0	0.2848	-2.4166	0	-3.3773	-3.5511	0	0	0	0.379	1.3625

WEEKEND (SI units)

EPRI														EXTENDED			
hour	a0	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	a11	a12	a13	no_dw	no_cw	senior	no_pay
	(const)	(per)	(age1)	(age2)	(age3)	(therm)	(tank)	(wtmp)	(atmp)	(athome)	(spring)	(summer)	(fall)	(winter)	[1]	[2]	[3]	[4]
11--6	0	0.3036	0	0	0	0	-0.0017	0	0	0	0.3433	0.6341	0	0.642	0	0	0.379	1.3625
6--8	3.8036	0	0	1.1515	1.3389	0	0.0269	0	-0.214	0	0	-5.9393	-2.4806	0	0	0	0.379	1.3625
8--11	0	0	1.3045	2.7796	5.6808	0	0	0	0	0	4.7598	0	0	5.1164	0	17.9456	0.379	1.3625
11--1	2.5923	0	0.9009	2.007	5.4181	-0.1833	0	0	-0.3291	0	0	-4.784	0	5.6039	0	0	0.379	1.3625
1--5	1.817	0	1.7303	1.7201	3.8043	-0.1445	0.0083	0	-0.2467	0	0	-3.5602	0	4.4501	0	0	0.379	1.3625
5--7	3.1616	0	2.1596	2.9693	3.8607	-0.1806	0	0	-0.3584	0	0	-5.6168	0	6.202	5.147	0	0.379	1.3625
7--9	3.5007	0	1.2038	4.9574	2.3144	0	0.0187	0	-0.1969	0	3.1275	-3.6961	0	0	5.147	0	0.379	1.3625
9--11	0	0	1.8965	3.0862	2.5695	0	0	0	0	0	0	0	-1.4339	0	0	0	0.379	1.3625

[1] no_dw is only applied to households without dishwashers, see no_dw equation

[2] no_cw is only applied to households without clothes washers, see no_cw equation

[3] senior has a value of 1 in households that are not seniors only.

[4] no_pay has a value of 1 in households that pay for the hot water they use.

2.3.4 Practical Implementation Issues in Water Heater

Due to uncertainty in the hot water consumption estimation given by equation (2.25), and due the approximation used for calculating α_{WH}, β_{WH} and γ_{WH} parameters, the water temperature may exceed the set limits. Therefore, it is safer to send the control signal to the lower heating element and let the upper element work without control as proposed in [17]. Normally, the water temperature is controlled by the lower heating element most of the time, whereas the upper element works few minutes a day during excessive hot water use.

It is also recommended to set the lower thermostat set point to a higher value than the upper solver set point limit, but lower than the safety switch limit. This will prevent interfering of the lower thermostat decisions with the control decisions for most of the times, and at the same time prevent the water temperature from reaching the safety switch set point. The upper thermostat should be set to a value that is slightly lower than the lower limit of the water temperature; this ensures that the water temperature will not go below a certain value in case of excessive hot water use. The modifications in the water heating elements wiring should be approved by the water heater manufacturer in order not to void the manufacturer warranty.

2.4 Refrigerator

Domestic refrigerators and freezers are common appliances which are available in almost every house. Refrigerators accounted for 8% of total residential electricity use in Ontario in 2007 [3], whereas freezers accounted for 2% only as shown in Figure 1.3. A domestic refrigerator consists mainly of the following parts: compressor, condenser, throttling valve and evaporator as shown in Figure 2.4. The refrigeration cycle consists mainly of four processes as shown in the pressure enthalpy diagram in Figure 2.5. For both figures, in process 2-3, the compressor consumes a power P_{in} (kW) to compress the refrigerant gas. This will increase the gas pressure and temperature by an isentropic (constant entropy) process. In this process, $P_{in} = m_r\,(h_3 - h_2)$, where m_r is the refrigerant mass flow rate (kg/s) and h is the refrigerant enthalpy (kJ/kg). In process 3-4, the condenser, which is mounted outside the refrigerator, helps to cool down the high pressure gas and to change it into liquid in a constant pressure process. In this process, $Q_{out} = m_r\,(h_4 - h_3)$, where Q_{out} is the amount of heat (kW) rejected by the condenser to the ambient, and m_r is the refrigerant mass flow rate. The throttling or expansion valve, in process 4-1, allows the refrigerant liquid to move from a high pressure zone into a low pressure zone in a constant enthalpy process $(h_4 = h_1)$, so it expands and evaporates and becomes very cold. The evaporator, which is the cold coil inside the refrigerator in process 1-2, allows the refrigerant to absorb heat by a constant pressure process, making the inside of the refrigerator cold. In this process $Q_{in} = m_r\,(h_4 - h_1)$, where Q_{in} is the amount of heat removed from the refrigerator cabinet.

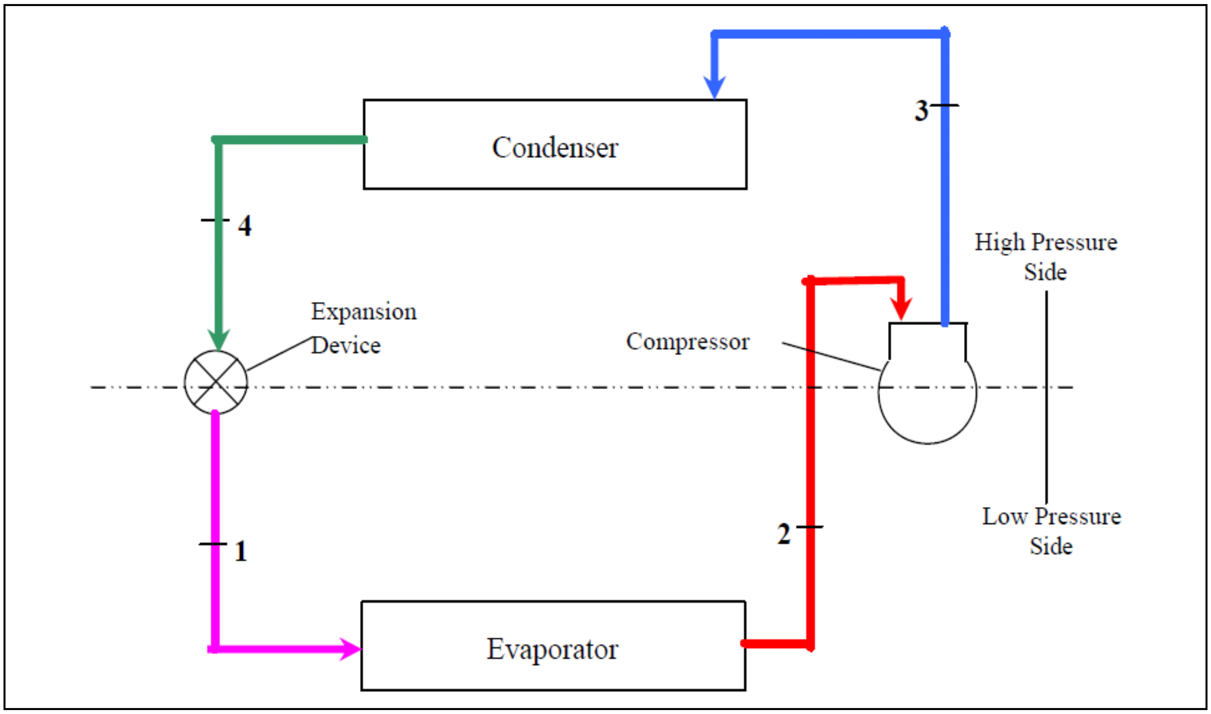


Figure 2.4. Components of a domestic refrigerator [18].

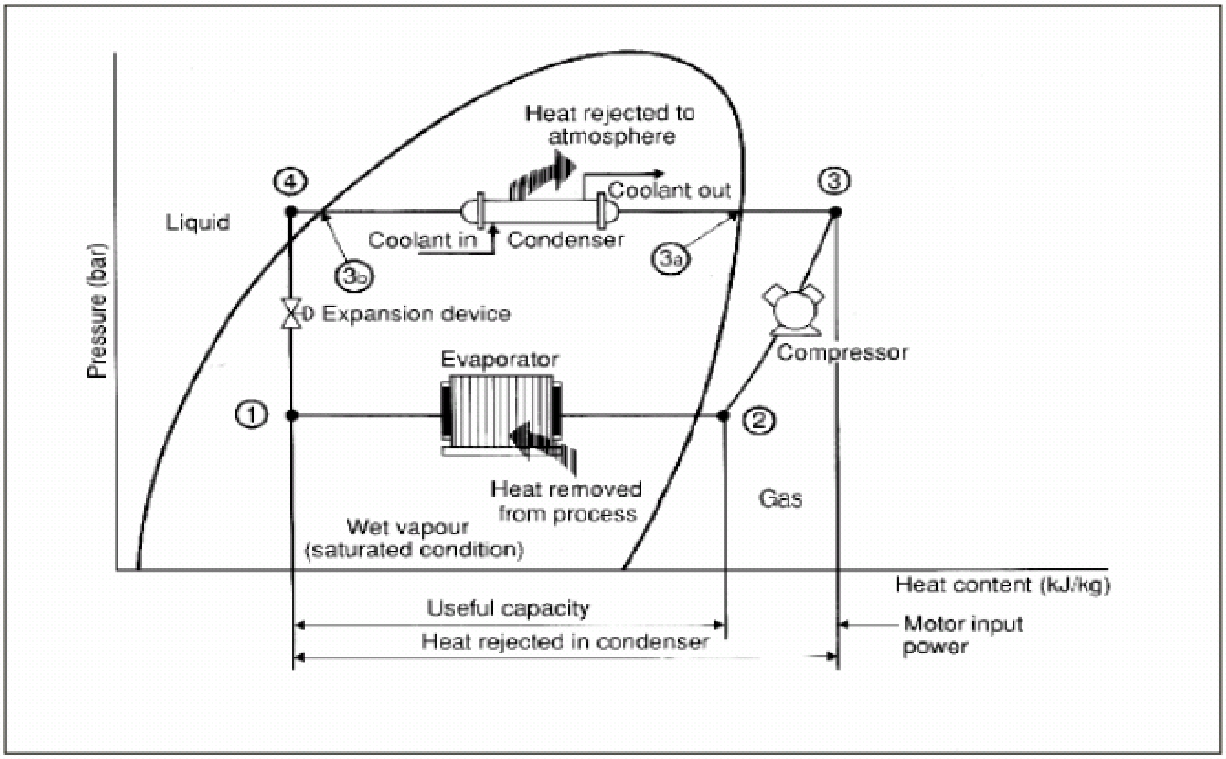


Figure 2.5. Pressure-Enthalpy diagram of a domestic refrigerator [18].

In refrigeration, a parameter of interest is the ratio of the amount of heat absorbed by the evaporator to the amount of energy consumed by the compressor, which is called the refrigerator coefficient of performance (COP), where ($COP = Q_{in}/P_{in}$). This coefficient has transient characteristics as shown in Figure 2.6 [19], where its value settles down after 2-3 hours; this means that the refrigerator is always working in the transient mode.

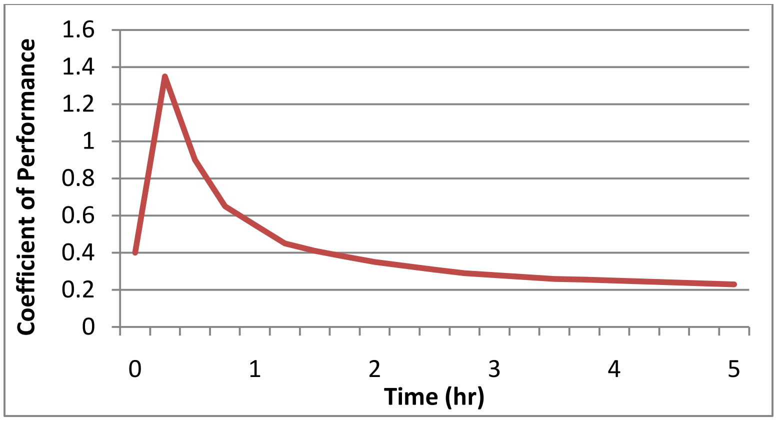


Figure 2.6. Change of the refrigerator COP with time [19].

When heat is removed from the humid air inside a closed cabinet, its dry bulb temperature cools down due to a process known as a sensible heat reduction, Q_{sen}. When air humidity is reduced due to condensation of vapor over the evaporator cold plate the involved heat loss in this process is called a latent heat loss, Q_{lat}, because energy is consumed to change only the phase of water vapor into water. Hence the total heat reduction is $Q_{in} = Q_{sen} + Q_{lat}$.

The sensible heat flow can be expressed in SI-units as $Q_{sen} = m\, C_p\, \Delta T$, where C_p = 1.005 is the specific heat capacity of air in kJ/kg.^{o}C, and $m = \rho\, V$, where ρ is the air density at standard conditions in kg/m^3 and V is the volume in m^3. The latent heat flow can be expressed in SI-units using $Q_{lat} = h_{we}\, \rho\, \Delta x$, where h_{we} = 2502 is the latent heat of vaporization of water in kJ/kg, ρ = 1.205 is the air density at standard conditions in kg/ m^3, and Δx is the humidity ratio difference (kg water/kg dry air).

2.4.1 Operational Constraints

The mathematical model of the refrigerator is represented in [9] by the following set of equations:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = FR \\ 0 & \text{if } t \notin T_i\,, i = FR \end{cases} \qquad EOT_i \;\le\; T_i \;\le\; LOT_i \tag{2.26}$$

$$S_i(t=1) = \begin{cases} 1 & \text{if } \theta_i(t=0) < \theta_i^{low} \\ 0 & \text{if } \theta_i(t=0) < \theta_i^{up} \end{cases} \qquad \forall t \,\in\, T_i\,, i = FR \tag{2.27}$$

$$\theta_i^{low} \le \theta_i(t) \le \theta_i^{up} \qquad \forall t \,\in\, T_i\,, i = FR \tag{2.28}$$

$$\theta_i(t) = \theta_i(t-1) + \beta_i\, AL_i(t) - \alpha_i\, S_i(t) + \gamma_i\, \big(1 - S_i(t)\big) \qquad \forall t \,\in\, T_i\,, i = FR \tag{2.29}$$

$$U_i(t) - D_i(t) = S_i(t) - S_i(t-1) \qquad \forall t \,\in\, T_i\,, i = FR \tag{2.30}$$

$$U_i(t) - D_i(t) \le 1 \qquad \forall t \,\in\, T_i\,, i = FR \tag{2.31}$$

$$\sum_{k=t}^{t+MUT_i} S_i(k) \ge\; MUT_i -\; LPN\big(1 - U_i(t)\big) \qquad \forall\, t \in T_i\,, i = FR \tag{2.32}$$

$$\sum_{k=t}^{t+MUT_i-\mathbf{1}} S_i(k) \le\; LPN\big(1 - D_i(t)\big) \qquad \forall t \,\in\, T_i\,, i = FR \tag{2.33}$$

where $AL_{FR}(t)$ is the refrigerator activity level during each time interval, β_{FR} is the effect of activity level on the refrigerator temperature (^{o}C per unit of AL), α_{FR} is the cooling effect of the ON state of the refrigerator (^{o}C per interval) and γ_{FR} is the warming effect of the OFF state of the refrigerator (^{o}C per interval).

Equations (2.26) to (2.33) have the same meanings as (2.3) to (2.10) for the water heater with HWU is replaced by $AL_{FR}(t)$. It is easier to implement Equation (2.29) if it is modified as follows:

$$\theta_i(t) = \theta_i(t-1) + \beta_i \, AL_i(t) - \alpha_i S_i(t) + \gamma_i \qquad \forall t \in T_i\,, i = FR \qquad (2.34)$$

In this case, γ_{FR} is physically measurable quantity that stands for temperature increase due to energy transfer associated with the difference between the refrigerator inside temperature and the surrounding air temperature. Note that as the time interval has been selected to be 15 minutes, and since there is no problem for the refrigerator compressor to change its state every 15 minutes (if necessary), one does not have to worry about the minimum ON/OFF times of the refrigerator states. For this reason (2.30) to (2.33) can be ignored. Therefore, the final mathematical model of the refrigerator that is going to be implemented is given by the following equations:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = FR \\ 0 & \text{if } t \notin T_i\,, i = FR \end{cases} \qquad EOT_i \le T_i \le LOT_i \qquad (2.35)$$

$$S_i(t=1) = \begin{cases} 1 & \text{if } \theta_i(t=0) < \theta_i^{low} \\ 0 & \text{if } \theta_i(t=0) > \theta_i^{up} \end{cases} \qquad \forall t \in T_i\,, i = FR \qquad (2.36)$$

$$\theta_i^{low} \le \theta_i(t) \le \theta_i^{up} \qquad \forall t \in T_i\,, i = FR \qquad (2.37)$$

$$\theta_i(t) = \theta_i(t-1) + \beta_i \, AL_i(t) - \alpha_i \, S_i(t) + \gamma_i \qquad \forall t \in T_i\,, i = FR \qquad (2.38)$$

2.4.2 Calculation of Parameters

Unlike the case of the water heater, it is not easy to calculate the refrigerator parameters using the equations of thermodynamics. This is because of the presence of humid air inside the refrigerator cabinet and the complicated sensible and latent heat transfer issues in this case. Furthermore, the refrigerator usually retains different kinds of food with different thermodynamic characteristics. Also, using thermodynamic equations to calculate the refrigerator parameters requires data on refrigerant mass, refrigerator coefficient of performance, evaporator dimensions, refrigerator shield thermal resistance data, etc. These data are normally not supplied by manufacturers like in the case of water heater. Besides, there are a wide diversity of refrigerators sizes, designs and types. Therefore, measuring refrigerator parameters is faster, easier and more accurate with the availability of low cost high precision data loggers.

Figure 2.7 shows an example of a temperature inside a refrigerator which was measured using a data logger. One can see that the refrigerator temperature rises by 0.081^{o}C/min. when the compressor is OFF. From this one can calculate γ_{FR} = 0.081^{o}C/min. × 15 min.= 1.21^{o}C for the case of 15-minutes time intervals. It is also observed from Figure 2.7 that when the compressor is ON, the temperature drops by 0.285^{o}C/minute. This is a result of temperature drop due to compressor operation minus temperature rise due to ambient heat entering the refrigerator through the shield and door gasket. Hence, the temperature drop due to compressor operation only is 0.081^{o}C/min. + 0.285^{o}C/min. = 0.366^{o}C/min. Accordingly α_{FR} = 0.366^{o}C/min. × 15 min.= 5.5^{o}C, for the case of 15-minutes time intervals.

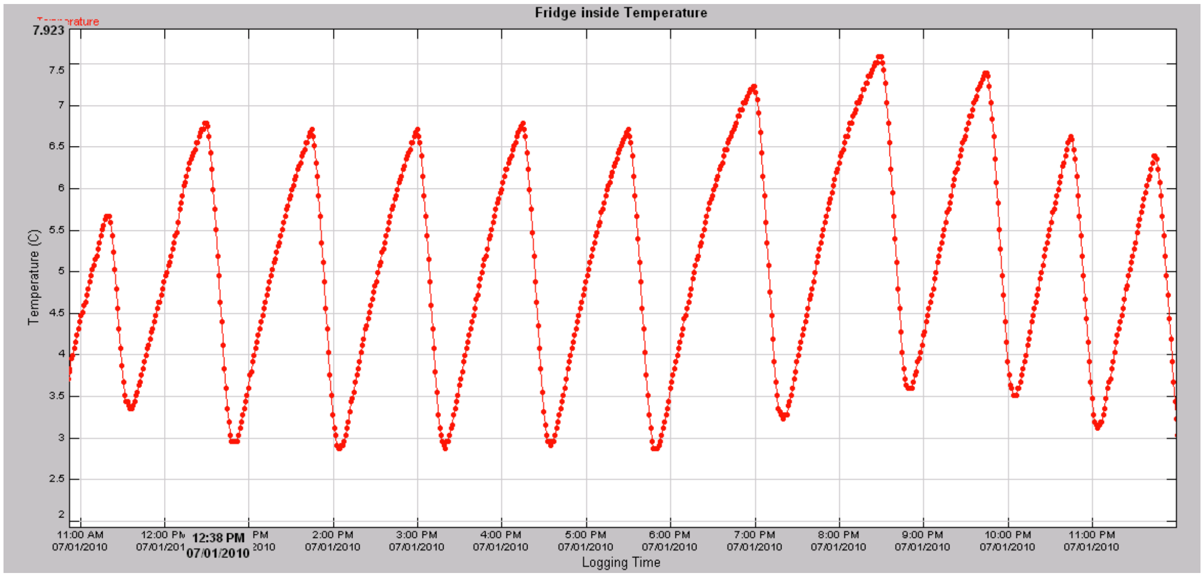


Figure 2.7. Refrigerator temperature logged events.

To calculate β_{FR}, one first needs to calculate the activity level $AL_{FR}(t)$; then, the term $\beta_{FR}\ AL_{FR}(t)$ will represent the temperature rise due to refrigerator usage such as door opening, fresh food loading and cold food off-loading, etc.

If one go back to Figure 2.1, periods of base-load electricity consumption can be noticed, which represents time periods of inactivity inside the house, when occupants are either sleeping or outside the house, and therefore the probability of the refrigerator door being opened is zero. By inspection, the value of this base-load consumption is approximately 50% of the average hourly electrical energy consumption. Therefore, to determine the refrigerator activity level it can be assumed that the base-load consumption is 50% of the average household consumption; thus, any load that is less than the base-load will not contribute to the refrigerator activity. In view of that, the refrigerator activity level can be calculated using the following equations:

$$EEC_1(t) = EEC(t) - \frac{0.5\,(\sum_{t=1}^{t=T} EEC(t))}{T} \quad , \qquad EEC_1(t) > 0 \tag{2.39}$$

$$AL_{FR}(t) = EEC_1(t) \div \sum_{t=1}^{t=T} EEC_1(t) \tag{2.40}$$

where $EEC(t)$ represents the household electrical energy consumption at time t, and $EEC_1(t)$ is a dummy function represents the amount of electrical energy consumption which is greater than the base-load energy consumption that was assumed to be equal to 50% of the daily average energy consumption. The resulting refrigerator activity level is shown in Figure 2.8.

Calculating β_{FR}, which represents the weight of that activity level on the refrigerator temperature increment, is a challenge, since very little research that examines how to find out the effect of door opening on refrigerator operation and energy consumption is available. Heat and mass transfer calculation during refrigerator door opening is very complex, the presence of shelves and food in the refrigerator make these calculations even more difficult. Therefore, it is challenging to find a common solution that can be applied to different refrigerators [20]. It was found that daily door openings for a family usage will give 20% to 32% energy consumption increase over the rated value [21]. The increase in refrigerator energy consumption is caused by heat gain from the light bulb, sensible and latent heat gain, fresh food loading or cold food unloading. It is assumed here that the extra daily energy consumption due to refrigerator door opening depends on a family size. Thus, a family of two people will be assumed to add 20% to the daily refrigerator consumption, and each extra family member is assumed to add an additional 2% due to refrigerator door opening, this is just an assumption that should be developed through more research, measurement and statistical calculations. Accordingly, if one considers the hourly energy consumption shown in Figure 2.1 to represent a family of four people, then the extra energy consumption due to the refrigerator door opening is 24%. Therefore, β_{FR} should be chosen in this case so that this extra energy consumption does not exceed 24% for the whole day. For example, It was found by using GLPK that β_{FR} = 0.28 for the case of four people using a refrigerator rated at 250 W.

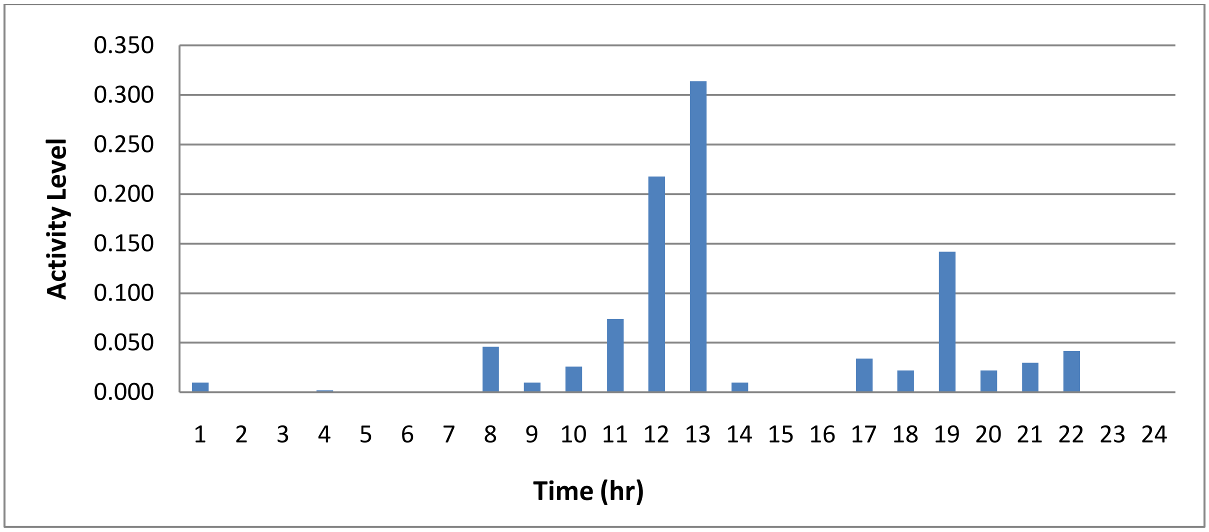


Figure 2.8. Refrigerator estimated door opening activity level.

It is important to highlight the fact that measuring the temperature change due to door opening does not help in estimating β_{FR} because when the door opens, the cool air will escape from the refrigerator and will be replaced by ambient hot air, but the fresh food temperature will gain temperature slowly as it has higher thermal mass than air. After closing the refrigerator door, the air temperature will start to cool down again even if the compressor is still in the OFF state due to heat exchange with the surrounding cold fresh food and refrigerator walls.

2.5 Air-conditioning and Heating

Residential space air cooling accounted for 14% of total residential electricity use in Ontario in 2007 [3], whereas space heating accounted for 30% as shown in Figure 1.3.

2.5.1 Operational Constraints

The mathematical model of the AC/Heating system is given by the following set of equations proposed in [9]:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = AC/H \\ 0 & \text{if } t \notin T_i\,, i = AC/H \end{cases} \qquad EOT_i \le T_i \le LOT_i \qquad (2.41)$$

$$S_i(t=1) = \begin{cases} 1 & \text{if } \theta_{in}(t=0) < \theta_{in}^{low} \\ 0 & \text{if } \theta_{in}(t=0) > \theta_{in}^{up} \end{cases} \qquad \forall t \in T_i\,, i = AC/H \qquad (2.42)$$

$$\theta_{in}^{low} \leq \theta_{in}(t) \leq \theta_{in}^{up} \qquad \forall t \in T_i, i = AC/H \tag{2.43}$$

$$\begin{aligned}\theta_{in}(t) = \theta_{in}(t-1) + \beta_{AC}\, AL(t) - \alpha_{AC}\, S_i(t) \\ + \gamma_{AC}\left(1 - S_i(t)\right) + \mu_{AC}\left(\theta_{out}(t) - \theta_{in}(t)\right) \qquad \forall t \in T_i, i = AC\end{aligned} \tag{2.44}$$

$$\begin{aligned}\theta_{in}(t) = \theta_{in}(t-1) + \beta_{H}\, AL(t) + \alpha_{H}\, S_i(t) \\ - \gamma_{H}\left(1 - S_i(t)\right) - \mu_{H}(\theta_{out}(t) - \theta_{in}(t)) \qquad \forall t \in T_i, i = H\end{aligned} \tag{2.45}$$

$$U_i(t) - D_i(t) = S_i(t) - S_i(t-1) \qquad \forall t \in T_i, i = AC/H \tag{2.46}$$

$$U_i(t) - D_i(t) \leq 1 \qquad \forall t \in T_i, i = AC/H \tag{2.47}$$

$$\sum_{k=t}^{t+MUT_i} S_i(k) \geq MUT_i - LPN\left(1 - U_i(t)\right) \qquad \forall\, t \in T_i, i = AC/H \tag{2.48}$$

$$\sum_{k=t}^{t+MUT_i-\mathbf{1}} S_i(k) \leq LPN\left(1 - D_i(t)\right) \qquad \forall t \in T_i, i = AC/H \tag{2.49}$$

where β_{AC} and β_H are the effect of Activity Level on the AC/Heating temperature (^{o}C per unit of Activity Level), α_{AC} and α_H are the cooling/heating effect of one (ON) state of AC/Heating (^{o}C per interval), γ_{AC} and γ_H are the cooling/warming effect of one (OFF) state of AC/H (^{o}C per interval), μ_{AC} and μ_H are the indoor/Outdoor heat transfer effect and AL is the Activity Level of the AC/Heating system.

A more practical approach is to combine equations (2.44) and (2.45) into one equation, as follows:

$$\begin{aligned}\theta_{in}(t) = \theta_{in}(t-1) + \beta_{AC}\, AL(t) - \alpha_{AC}\, S_{AC}(t) + \alpha_{H}\, S_{H}(t) \\ +\gamma_{AC}\left(\theta_{out}(t) - \theta_{in}(t)\right) \qquad \forall t \in T_i, i = AC/H\end{aligned} \tag{2.50}$$

Where the parameter γ_{AC} is redefined as representing the effect of energy losses on room temperature associated with the indoor and outdoor temperature differences. Another constraint could be added to ensure that the AC and the heating systems are not operational at the same time. Moreover, as the time period has been selected to be 15-minutes, and since there is no problem for the AC compressor or heating element to change its state every 15-minutes (if necessary), one doesn't have to worry about the minimum ON/OFF times of the cooling/heating states; for this reason (2.46) to (2.49) can be ignored. Therefore, the mathematical model of the cooling/heating appliance is represented by the following set of equations:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = AC/H \\ 0 & \text{if } t \notin T_i\,, i = AC/H \end{cases} \qquad EOT_i \le T_i \le LOT_i \tag{2.51}$$

$$S_i(t=1) = \begin{cases} 1 & \text{if } \theta_{in}(t=0) < \theta_{in}^{low} \\ 0 & \text{if } \theta_{in}(t=0) > \theta_{in}^{up} \end{cases} \qquad \forall t \in T_i\,, i = AC/H \tag{2.52}$$

$$\theta_{in}^{low} \le \theta_{in}(t) \le \theta_{in}^{up} \qquad \forall t \in T_i\,, i = AC/H \tag{2.53}$$

$$\begin{aligned} \theta_{in}(t) = \theta_{in}(t-1) + \beta_{AC}\, AL(t) - \alpha_{AC}\, S_{AC}(t) + \alpha_H\, S_H(t) \\ + \gamma_{AC}\big(\theta_{out}(t) - \theta_{in}(t)\big) \qquad \forall t \in T_i\,, i = AC/H \end{aligned} \tag{2.54}$$

$$S_{AC}(t) + S_H(t) \le 1 \qquad \forall t \in T_i\,, i = AC/H \tag{2.55}$$

2.5.2 Calculation of Parameters

The parameters α_{AC}, α_H and γ_{AC} can be found from measurements as in the case of the refrigerator. It was found from measurements in a certain house for example, when the space heating thermostat set-point moved back from 23°C to 20°C for 2 hours, the house temperature dropped by 1.8°C as shown in Figure 2.9. Knowing that the indoor/outdoor temperature difference at that time was roughly 30°C, γ_{AC} was calculated from (2.54) after setting α_{AC}, β_{AC} and α_H to zero. In this case value of $\gamma_{AC} = \frac{1.8}{30 \times 2 \times 4} = 0.0075$ for 15-minute time intervals where 4 is the number of 15-minute intervals in one hour. When a space heating thermostat set-point moved up from 20°C to 25°C, it was noticed that the house temperature increased by 2.41°C after 45 minutes. Using the value of γ_{AC} obtained in the previous step and (2.54) again, it can be shown that α_H = 1.02 °C, for 15-minute time intervals. The house space heater rating was 21 kW. This means that α_H will change by 1.02/21 °C for each kW change of heating energy. If the house is going to be cooled in summer by a central AC system rated at 11.5 kW, then α_{AC} should be equal to 0.51.

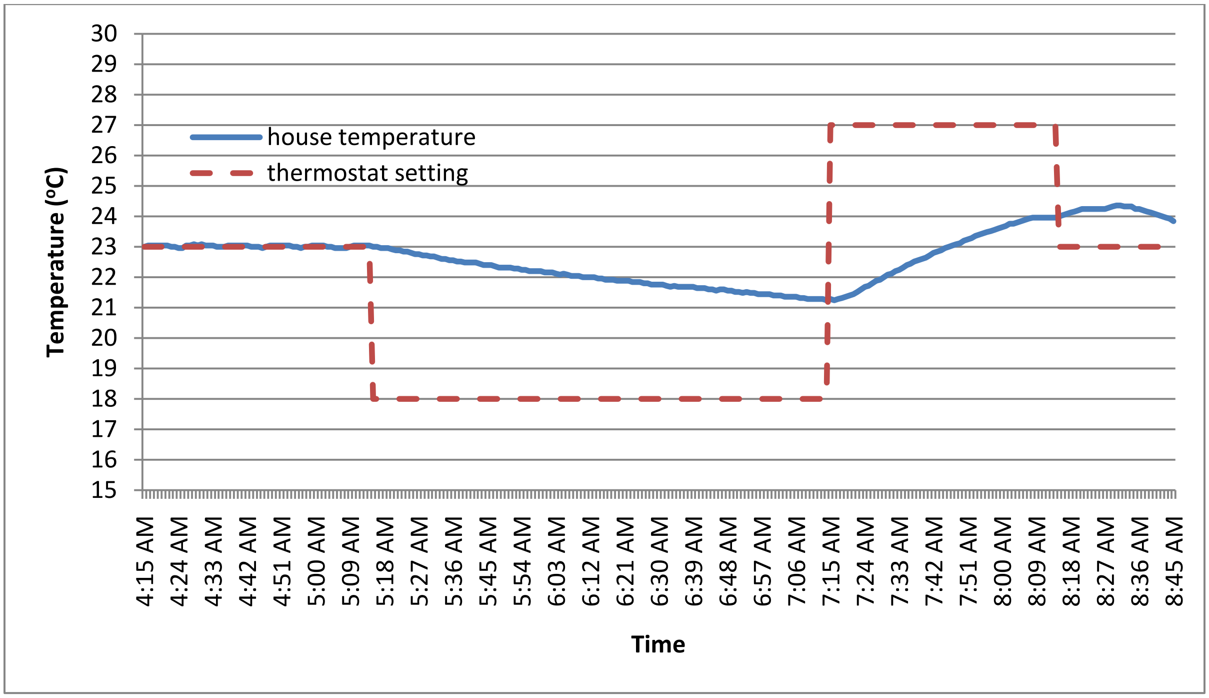


Figure 2.9. Inside temperature of a house after changing the thermostat set point.

Different household occupant activities have different effects on space heating/cooling consumption. When many persons are present inside a house, their bodies emit energy that contributes to the house heating. Many persons inside a house may also lead to many electrical appliances being used at the same time and each appliance will add its electrical power consumption inside a house in a different way. For example, when a 2000 watts stove heating element is working, it adds all that power to the space heating power to increase the house temperature. However, when the fan above the stove is working the house will lose energy along with the hot air that is leaving the house through that fan. Calculating β_{AC} requires detailed information about the existing appliances in the house, their energy consumption and probability of their use at each time interval. Therefore, each house should be studied in detail, separately. But, the common thing in residential space heating/cooling is that the base-load electricity consumption contributes to the house heating, therefore, the activity level $AL(t)$, shown in Figure 2.2, will be used for heating/cooling appliances.

Since different appliances contribute differently to house heating, as was mentioned before, It can be assumed that 50% of the hourly electrical consumption will eventually be converted into heat and contribute to the house temperature increment [22]; Hence β_{AC} , for our example can be selected so that its value will make a saving of 50% in the heating energy consumption for the whole day. Using GLPK it was found that β_{AC} is equal to 0.0275.

2.6 Energy Storage Devices

There is increased interest in the deployment of energy generated from renewable resources such as small scale photo voltaic (PV) systems, wind turbines, etc. The output power generated from theses sources is intermittent, unreliable, and dependent on weather conditions, physical locations and time. This kind of power is also not dispatchable and it could be available at periods of low demand; therefore, it is usually integrated with some storage devices, such as batteries, which can store the generated energy for a certain period of time and release it when demand increases. However, besides being expensive, batteries have limited capacities; thus, if there is a surplus of electricity produced by, for example, a domestic PV system, then this extra power should be either dissipated into a dissipation resistance or injected into the local grid if allowed.

2.6.1 Operational Constraints

The mathematical model of the ESD is given by the following set of equations proposed in [9]:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i, i = ESD \\ 0 & \text{if } t \notin T_i, i = ESD \end{cases} \qquad EOT_i \le T_i \le LOT_i \tag{2.56}$$

$$ESL_i(t) \ge ESL_i^{min} \qquad \forall t \in T_i, i = ESD \tag{2.57}$$

$$ESL_i(t) = ESL_i(t-1) - S_i(t)\, Discharg_i + Charge_i(t) \qquad \forall t \in T_i, i = ESD \tag{2.58}$$

$$U_i(t) - D_i(t) = S_i(t) - S_i(t-1) \qquad \forall t \in T_i, i = ESD \tag{2.59}$$

$$U_i(t) - D_i(t) \le 1 \qquad \forall t \in T_i, i = ESD \tag{2.60}$$

$$\sum_{k=t}^{t+MUT_i} S_i(k) \ge MUT_i - LPN\big(1 - U_i(t)\big) \qquad \forall t \in T_i, i = ESD \tag{2.61}$$

$$\sum_{k=t}^{t+MUT_i-\mathbf{1}} S_i(k) \le LPN\big(1 - D_i(t)\big) \qquad \forall t \in T_i, i = ESD \tag{2.62}$$

where $Discharg_{ESD}$ is the ESD discharge power, $Charge_{ESD}(t)$ is the charge power during each time interval, and $ESL_{ESD}(t)$ is the storage level that should not go below a minimum value ESL_{ESD}^{min}.

Since intermittent charge/discharge is allowed, equations (2.59) to (2.62) for minimum up/down time can be ignored.

There are no parameters to be calculated in the ESD mathematical model. $Discharg_{ESD}$ and ESL_{ESD}^{min} can be found from the ESD system data sheets and devices' power ratings.

2.6.2 Domestic PV system modeling

Figure 2.10 shows one method of a grid connected domestic PV electric power system in which the DC/DC converter can be in two operational modes, the converter mode to charge the battery with a limited power P_{CH} as recommended by the battery manufacturer and the inverter mode to discharge the battery's stored energy back to the system. The discharge power rating P_{DR} is determined by the DC/DC converter power rating. The AC power generated by the DC/AC inverter is consumed by the house appliances or injected to the utility grid in case of low house electric demand.

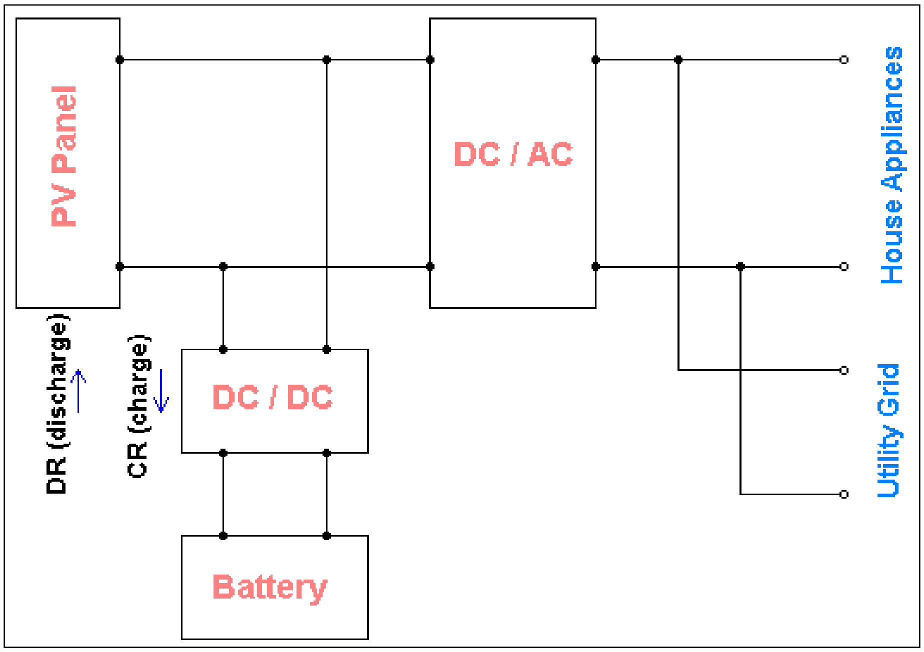


Figure 2.10. Domestic PV electric power system diagram.

The mathematical model of the PV system is as follows:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = DR \text{ or } CR \\ 0 & \text{if } t \notin T_i\,, i = DR \text{ or } CR \end{cases} \qquad EOT_i \le T_i \le LOT_i \qquad (2.63)$$

$$P_{CHR}(t) = \begin{cases} P_{CH} & \text{if} \quad Ppv(t) \geq P_{CH} \\ Ppv(t) & \text{if} \quad Ppv(t) \leq P_{CH} \end{cases} \tag{2.64}$$

$$BATESL^{min} \leq BATESL(t) \leq BATESL^{max} \tag{2.65}$$

$$BATESL(t) = BATESL(t-1) - S_{DR}(t)\, P_{DR} + S_{CR}(t)\, P_{CHR}(t) \tag{2.66}$$

$$S_{DR}(t) + S_{CR}(t) \leq 1 \tag{2.67}$$

where $S_{CR}(t)$ and $S_{DR}(t)$ are the battery charging/discharging binary decisions of the DC/DC converter, $BATESL(t)$ is the battery energy storage level at each time interval, $BATESL^{min}$ and $BATESL^{max}$ are the minimum and maximum battery energy storage level respectively, and $Ppv(t)$ is the power generated by the PV panel, as shown in Figure 2.11.

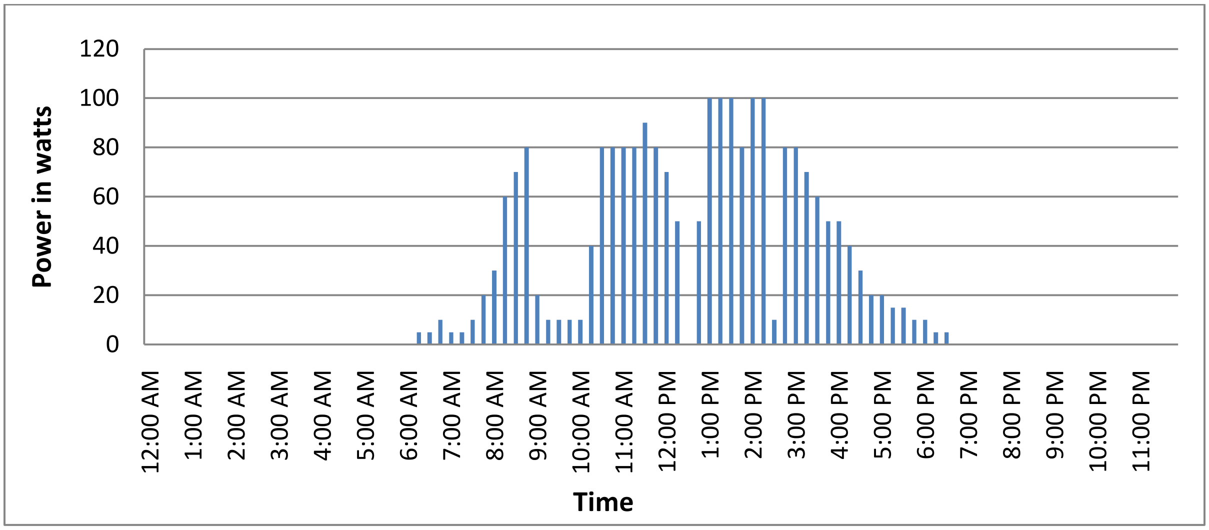


Figure 2.11. An example of power generated by a domestic solar system [12].

Equation (2.64) simulates the constant current battery charger operation which is normally used to charge the PV systems batteries. For simplicity it was assumed that the battery voltage is constant during the discharging/charging operations; thus, a constant current battery charging is assumed to be a constant power charging process. Equation (2.65) is to protect the battery against deep discharging and over charging. Equation (2.66) shows the effect of the charge/discharge decisions on the battery storage level, and equation (2.67) reflects the fact that the DC/DC converter does not operate in charge and discharge mode simultaneously in this particular configuration; however, this constraint can be ignored if separate charging and discharging units are used. In the context of TOU energy price scheme the optimizer will try to find the optimum charge/discharge binary decisions to get the maximum benefit from the PV system via charging the battery during periods of cheap energy prices, and discharging the battery to use or sell its stored energy during periods of expensive energy prices. To achieve this goal the objective function should also be modified as follows:

$$\begin{aligned} min\ J = \sum_{t\in TA}\Bigg[& \sum_{i\in A, i\neq LI, i\neq PV} C_t(t)P_iS_i(t) + \sum_{i=LI} C_t(t)P_iIL(t) - \sum_{i=DIS} C_t(t)P_iS_i(t) \\ & - \sum_{i=CHR} C_t(t)(Ppv(t) - P_iS_i(t))\Bigg] + \sum_{t\in TB}\Bigg[\sum_{i\in B} C_t(t)P_iS_i(t)\Bigg] \end{aligned} \tag{2.68}$$

It was assumed in the above model, that the DC/AC and DC/DC conversion efficiency is 100% and the customer sells the excess PV power to the utility at the same TOU rate $C_t(t)$. Using the above objective function yields optimum charging/discharging decisions, charging the battery from the PV source at time of cheap electricity prices, and discharging the stored battery energy to the grid at time of high electricity prices in order to maximize the financial benefits.

If the utility pays an incentive for the total domestic PV generation (whether it is consumed inside the house or injected into the grid), then this should be reflected in the objective function equation as follows:

$$\begin{aligned} min\ J = \sum_{t\in TA}\Bigg[& \sum_{i\in A, i\neq LI, i\neq PV} C_t(t)P_iS_i(t) + \sum_{i=LI} C_t(t)P_iIL(t) - \sum_{i=DIS} C_i(t)P_iS_i(t) \\ & - \sum_{i=CHR} C_i(t)(Ppv(t) - P_iS_i(t))\Bigg] + \sum_{t\in TB}\Bigg[\sum_{i\in B} C_t(t)P_iS_i(t)\Bigg] \end{aligned} \tag{2.69}$$

where $C_i(t)$ is the selling price of the PV generation. However, if the utility pays incentives for the surplus PV generation that is injected to the grid only, then objective function will be more complicated; this case is not studied here.

2.7 Dishwasher

Dishwashers represent a small component of residential energy consumption. They contributed approximately 0.5% of the total electrical energy used in Ontario in 2007 [3]. However, dishwashers consume high power during short period of times, which makes them relevant for peak demand programs. Moreover, a dishwasher's time of use can be shifted with satisfactory degree of customer's acceptance.

Dishwashing process is controlled by a step timer or an electronic control device, and its operation lasts between about 15 minutes and up to 2 hours, depending on the program and temperature chosen [23]. Electrical energy is used mainly for heating up the water if it was not hot, drying the dishes by blowing hot air and driving the circulation pump motor. A dishwasher's power demand profile may vary between machines and from program to program for the same machine. The dishwasher average power demand curves are given in Figure 2.12 for 15 minutes resolution and one hour resolution. The average energy consumption is roughly 1.2 kWh per cycle [23].

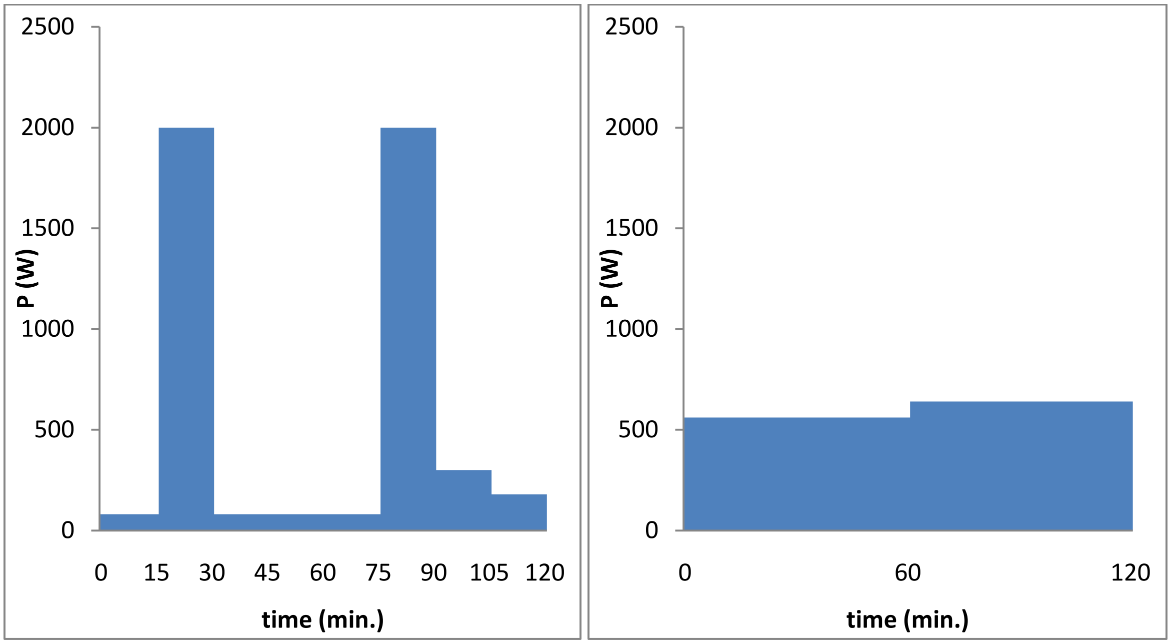


Figure 2.12. An example of a dishwasher power demand profile [23].

2.7.1 Operational Constraints

The mathematical model of the dishwasher is given by the following set of equations in [9]:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = DW \\ 0 & \text{if } t \notin T_i\,, i = DW \end{cases} \qquad EOT_i \leq T_i \leq LOT_i \qquad (2.70)$$

$$SL_i(t) = SL_i(t-1) + \beta_i\, AL(t) - S_i(t)\, Cap_i \qquad \forall t \in T_i\,, i = DW \qquad (2.71)$$

$$S_i(t) = 0 \text{ if } SL_i < Cap_i \qquad \forall t \in T_i, i = DW \qquad (2.72)$$

$$\sum_{k=t}^{LOT_i} S_i(k) \geq 1 \text{ if } SL_i(t) \geq Cap_i \qquad \forall t \in T_i, i = DW \qquad (2.73)$$

where β_{DW}, is the effect of the Activity Level on the storage level of dishwasher. Constraint (2.70) specifies the time period over which the customer can put the dishwasher into operation. In (2.71), the storage level of the dishwasher at time t, $SL_{DW}(t)$, is a function of the previous hour storage level, customer activity level and ON/OFF state of the dishwasher at time t. Constraint (2.72) ensures that when the storage level of the dishwasher is less than the dishwasher capacity Cap_{DW}, it cannot start to operate. Constraint (2.73) ensures that if the storage level exceeded the dishwasher capacity at a certain hour, then the dishwasher should be switched ON at least once after that hour but before the last operating hour LOT_i of the dishwasher.

Constraints (2.72) and (2.73) are non-linear; therefore, they were linearized in [9] by replacing them with an appropriate set of linear equations. The storage level, $SL_{DW}(t)$, was assigned energy units in [9] to represent the energy required to clean the accumulated dirty dishes; however, it can also be given units of kg to represent the mass of dirty dishes, or it can be just an integer to represent the number of dirty dishes. It is probably more convenient to use a positive integer between 0 and 100 to represent the percent of dirty dishes with respect to the dishwasher capacity.

Due to the complexity of the dishwasher equations, its time schedule has been chosen to be 24 one-hour time intervals instead of 96 fifteen-minute intervals, in order to reduce the computational burden. One hour time intervals for the dishwasher is adequate, since some of its operating programming modes require at least two hours to finish; therefore, a minimum ON time of 2 hours should be used for this appliance.

The non-linear equations (2.72) and (2.73) can be avoided if a different dishwasher operation algorithm is chosen. For example, it can be stated that the dishwasher should operate so that the number of dirty dishes by the end of the day should not exceed 90% of the dishwasher capacity which can be simply modeled by the following linear equation:

$$SL_i(24) \leq 0.9\ Cap_i \qquad \forall t \in T_i, i = DW \qquad (2.74)$$

This means that the dishwasher storage level at hour 24, which is the last hour of the day, should be equal to or less than 90% of the storage capacity. Equation (2.74) was used instead of (2.72) and (2.73) in the dishwasher model here.

2.7.2 Calculation of Parameters

It can be expected that during base-load periods in the household, no dishes would be used or loaded to the dishwasher. Thus, the dishwasher activity level data can be assumed to be similar to that of the refrigerator activity level given in Figure 2.8.

To calculate β_{DW}, one can assume that the dishwasher should be used at least once by the end of the day, in case of a regular house activity level. Therefore, β_{DW} should be selected so that the dishwasher storage capacity is reached between 7:00 PM and 9:00 PM. However, the difficulty lies in estimating the initial storage capacity which represents the number of dishes that were not washed from the previous day. In this case further simplifications to the dishwasher mathematical model could be achieved by assuming that the dishwasher should operate for maximum of 2 successive hours per day if required. The resulting mathematical equation for this model is given by the following set of equations:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = DW \\ 0 & \text{if } t \notin T_i\,, i = DW \end{cases} \qquad EOT_i \le T_i \le LOT_i \tag{2.75}$$

$$U_i(t) - D_i(t) = S_i(t) - S_i(t-1) \qquad \forall t \in T_i\,, i = DW \tag{2.76}$$

$$U_i(t) - D_i(t) \le 1 \qquad \forall t \in T_i\,, i = DW \tag{2.77}$$

$$\sum_{t \in T_i} S_i(k) = ROT_i \qquad \forall\, t \in T_i\,, i = DW \tag{2.78}$$

$$\sum_{k=t-1}^{t-MUT_i+1} U_i(k) \le S_i(t) \qquad \forall\, t \in T_i\,, i = DW \tag{2.79}$$

$$\sum_{k=t-1}^{t+MDT_i+\mathbf{1}} D_i(k) \le (1 - S_i(t)) \qquad \forall t \in T_i\,, i = DW \tag{2.80}$$

This simplification in dishwasher equations makes the solution much faster. It also enables the use of 96 fifteen-minute time intervals for the dishwasher, like other appliances. Using identical and synchronized set of time schedules for all appliances makes implementation faster and more realistic.

2.7.3 Practical implementation issues

It should be noted that the dishwasher optimum ON/OFF decisions $S_{DW}(t)$ generated by the EHMS will not be used to directly trigger the dishwasher; it will either be used to trigger a pilot lamp to inform the customer about the optimum times for running the dishwasher, or to activate the dishwasher input supply socket. In both cases, it is the customer who will decide to run the dishwasher or not during these proposed times. In case of using an actuator to activate or deactivate the dishwasher input supply socket, it should be equipped with an override bottom to enable the customer to ignore the EHMS decision if it was OFF and run the dishwasher when desired. A signal should be sent to the EHMS in this case to inform that the EHMS optimum decision has been disregarded in order to re-run the optimization and calculate and send the new optimum decisions to all other appliances.

2.8 Cloth Washer

Cloth washers like dishwashers represent a small component of residential energy consumption. They consumed 0.7 PJ of electrical energy in Ontario in 2007, which is only 0.5% of total residential electrical energy consumption [3]. However, cloth washers consume high power during short periods of time, making them relevant for peak-demand management programs. Moreover, a washer’s time of use can be shifted with a high degree of customer’s acceptance.

The cloth washing process is controlled by a step timer or an electronic control device and lasts between about 15 minutes and 2 hours, depending on the washing program chosen [23]. Electrical energy is used mainly for driving the drum motor and heating up the water, if it was not hot enough; in spite of the fact that about 3/4 to 2/3 of the water is used as cold water for rinsing. An example of the washer average power demand profile is given in Figure 2.13 for 15 minutes resolution [23]. The average energy consumption in this case is roughly 0.9 kWh per cycle.

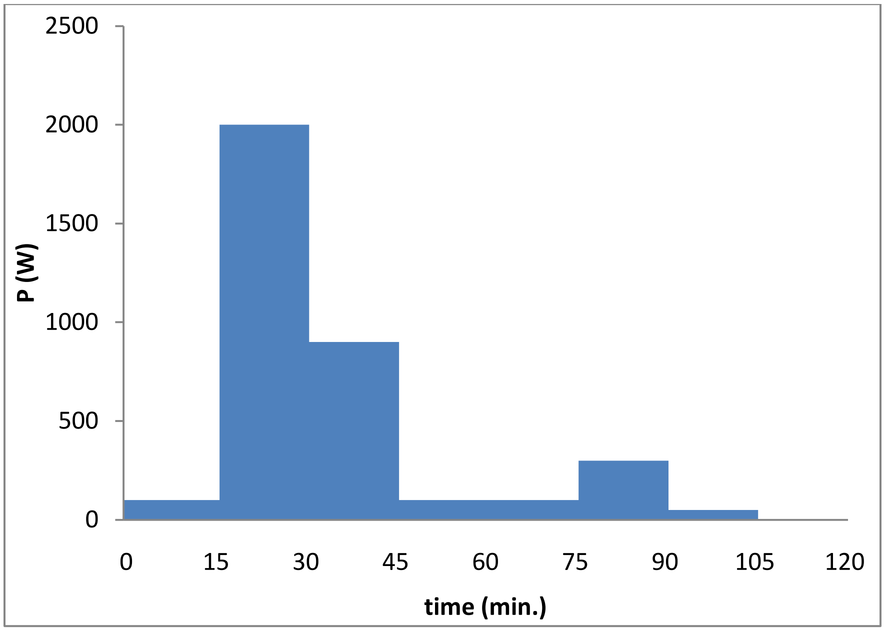


Figure 2.13. Cloth washer electrical energy consumption profile [23].

2.8.1 Operational Constraints

The mathematical model of the washer is given by the following set of equations [9]:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = W \\ 0 & \text{if } t \notin T_i\,, i = W \end{cases} \qquad EOT_i \ \le \ T_i \ \le \ LOT_i \qquad (2.81)$$

$$SL_i(t) = SL_i(t-1) + \beta_i\, AL(t) - S_i(t)\, Cap_i \qquad \forall t \in T_i\,, i = W \qquad (2.82)$$

$$S_i(t) = 0 \text{ if } SL_i \ < \ Cap_i \qquad \forall t \in T_i\,, i = W \qquad (2.83)$$

$$\sum_{k=t}^{LOT_i} S_i(k) \ \ge \ 1 \text{ if } SL_i\,(t) \ \ge \ Cap_i \qquad \forall t \in T_i\,, i = W \qquad (2.84)$$

where β_W is the effect of the Activity Level on the storage level of washer. Constraint (2.81) specifies the time period over which the customer can put the washer into operation. In (2.82), the storage level of the washer at time $t, SL_W(t)$, is a function of the previous hour's storage level, customer activity level and ON/OFF state of the washer at time t. Constraint (2.83) ensures that when storage level of the washer is less than the washer's capacity Cap_W, it cannot start to operate. Constraint (2.84) means that if the storage level exceeded the washer's capacity at a certain hour, then the washer should be switched ON once, at least, after that hour but before the last operating hour LOT_W of the washer [9].

Constraints (2.83) and (2.84) are non-linear; therefore, they were linearized in [9] by replacing them with an appropriate set of linear equations. The storage level $SL_W(t)$ was given energy units in [9]; however, it can be given units of kg to represent the mass of accumulated dirty clothes to be washed. It is probably more convenient to use a positive integer between 0 and 100 for $SL_W(t)$ to represent the percent of unclean clothes with respect to the washer capacity.

Due to the complexity and non-linearity of washer equations, its time schedule has been chosen to be 24 one-hour time intervals, instead of 96 fifteen-minute intervals, in order to reduce the computational burden. One hour time interval for a washer is adequate since some of its operating programming modes require at least 2 hours; therefore, a minimum ON time of 2 hours should be used for this appliance. Care should be taken when calculating the cost of energy consumption of the washer by using the average energy consumption for the whole 2 hours.

The non-linear equations (2.83) and (2.84) can be avoided if a different washer operation algorithm is chosen. For example, it can be stated that the washer should operate so that the amount of unclean clothes by the end of the day does not exceed 90% of the washer capacity, which can be simply modeled by the following linear equation:

$$SL_i(24) \;\leq\; 0.9\ Cap_i \qquad \forall t\ \in\ T_i\,, i = W \qquad (2.85)$$

This means that the washer storage level at hour 24, the last hour of the day, should be equal to or less than 90% of the storage capacity. This cloth washer model is used here.

2.8.2 Calculations of Parameters

The effect of activity level on the washer use is shown in equation (2.82). However, the rate of unclean cloth accumulation is proportional to both indoor and outdoor residence activities and therefore household hourly energy consumption cannot be used to forecast outdoor residence activities. Hence, storage level of unclean cloth should be related to the number of persons living in the house instead of the activity level as shown in equation (2.86):

$$SL_i(t) = SL_i(t-1) + \left(\frac{100}{24\ \times 16}\right) person - \ S_i(t)\ Cap_i \qquad \forall t\ \in\ T_i\,, i = W \qquad (2.86)$$

where $person$ denotes the number of persons in the house, 24 is the number of hours per day, and 16 is the number of days a single person can fill the cloth washer capacity, which depends on the washer capacity and the person activity. The number 16 was chosen as an example for a family of 4 persons that washes their clothes every 4 days.

The difficulty here lies in estimating the initial storage capacity, which represents the number of clothes that were not washed from the previous day. Further simplifications to the washer mathematical model could be achieved by assuming that the washer should operate for a maximum of two successive hours. The resulting mathematical model in this case is similar to the constraints (2.75) to (2.80) of the dishwasher. This simplification in washer equations makes the solution faster. It also enables the use of 96 fifteen-minute time intervals for the washer, like other appliances. Using identical and synchronized set of time schedules for all appliances makes implementation faster and more realistic.

2.8.3 Practical Implementation Issues

It should be noted that the washer optimum ON/OFF decisions $S_W(t)$ generated by the EHMS will not be used to directly trigger the washer; it will either be used to trigger a pilot lamp to inform the customer that this is the optimum time for starting the washer, or to activate the washer input supply socket. In both cases, it is the customer who will decide to run the washer or not at this time. In case of using an actuator to activate or deactivate the washer input supply socket, it should be equipped with an override bottom to enable the customer to ignore the EHMS OFF decisions and run the washer when desired. A signal should be sent to the EHMS in this case to inform that the EHMS optimum decision has been ignored in order to re-run the optimization and calculate and send the new optimum decisions to all other appliances.

2.9 Dryer

Dryers consumed 11.4 PJ of energy in Ontario in 2007. Approximately 96% of dryers in Ontario use electricity to heat the drying air while 4% only use natural gas [3]. Electrical dyers contributed 7% of the total residential electrical energy consumption [3]; therefore, they are relevant for peak-demand management programs. Moreover, a dryer’s time of use can be shifted with a high degree of customers’ acceptance.

In dryers, energy is mainly needed to evaporate the water from the laundry, rotate the drum and drive the fan [23]. Normal power for the heating devices used for water evaporation is in the range of

2000 to 2500 W. The drying process is controlled either by a timer-function to determine the drying time or by a humidity controller to decide final humidity of the load [23]. Time delay functions are integrated in some dryers to shift the starting time by a certain number of hours. An example of a dryer consumption pattern is shown in Figure 2.14.

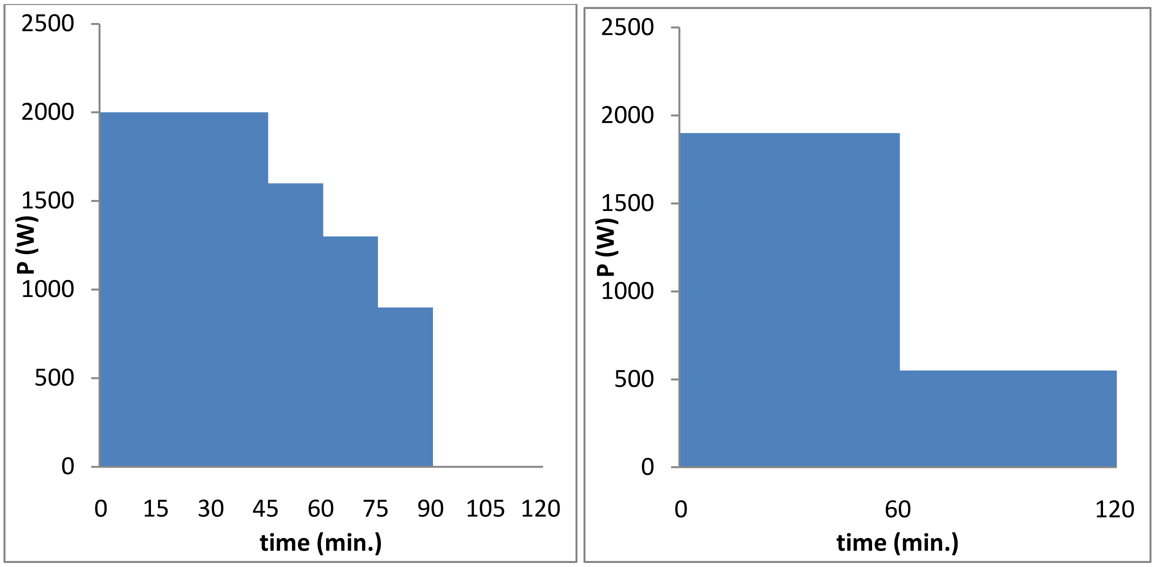


Figure 2.14. Dryer consumption pattern for 15 minutes and one hour resolution [23].

2.9.1 Operational Constraints

The following sets of equations were proposed in [9] for the dyer mathematical model:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i, i = DRY \\ 0 & \text{if } t \notin T_i, i = DRY \end{cases} \qquad EOT_i \leq T_i \leq LOT_i \qquad (2.87)$$

$$SL_i(t) = SL_i(t-1) + \beta_i\, AL(t) - S_i(t)\, Cap_i \qquad \forall t \in T_i, i = DRY \qquad (2.88)$$

$$S_i(t) \leq \sum_{k=1}^{MAT_{GAP}} S_W(t-k) \qquad \forall t \in T_i, i = DRY \qquad (2.89)$$

$$S_i(t) + S_W(t) \leq 1 \qquad \forall t \in T_i, i = DRY \qquad (2.90)$$

$$\sum_{t \in T_i} S_i(t) = \sum_{t \in T_W} S_W(t) \qquad \forall t \in T_i, i = DRY \qquad (2.91)$$

The first two equations are similar to those used for the washer. The last three equations are added to ensure the dyer is operating after the washer within a time gap that is not exceeding a certain value MAT_{GAP} [9].

From the implementation point of view, Equation (2.88) can be modified as follows:

$$SL_i(t) = SL_i(t-1) + S_W(t)\, Cap_W - S_i(t)\, Cap_i \qquad \forall t \in T_i, i = DRY \qquad (2.92)$$

This equation ensures that the dryer is loaded with clothes after washing is done. The storage level $SL_{DRY}(t)$ was assigned an integer number between 0 and 100 to represent the percent of wet cloth with respect to the dryer capacity. Because some of the dryer's operating programming modes require at least 2 hours to finish, a minimum ON time of 2 hours should be used for this appliance. However, care should be taken when calculating the cost of energy consumption of the dryer by using the average energy consumption per hour.

Similar to the dishwasher and cloth washer, the mathematical model for a dryer can be further simplified by allowing the dryer to operate for a maximum of two successive hours per day. In this case, the mathematical equations will be similar to equations (2.75) to (2.80) of the dishwasher.

2.9.2 Practical Implementation Issues

The dyer optimum ON/OFF decisions $S_{DRY}(t)$, generated by the EHMS, should not be used to directly trigger the dryer; it will be either used to trigger a pilot lamp to inform the customer about optimum time for starting the dryer or to activate the dryer input supply socket. In both cases, it is the customer who will decide when to run the dryer. In case of using an actuator to activate or deactivate the dryer input supply socket, it should be equipped with an override bottom to enable the customer to bypass the EHMS OFF decisions and run the dryer. A signal should be sent to the EHMS in this case in order to re-run the optimization and calculate and send the new optimum decisions to other appliances.

2.10 Stoves (Ranges)

Stoves, ranges and ovens operation are similar; therefore, they can be modeled using the same equations. With 87% of ranges working on electricity and remaining on natural gas, their share in total electrical energy consumption was approximately 6% in Ontario in 2007 [3]. Figure 2.15 shows an example of the daily power demand of an electric stove [23].

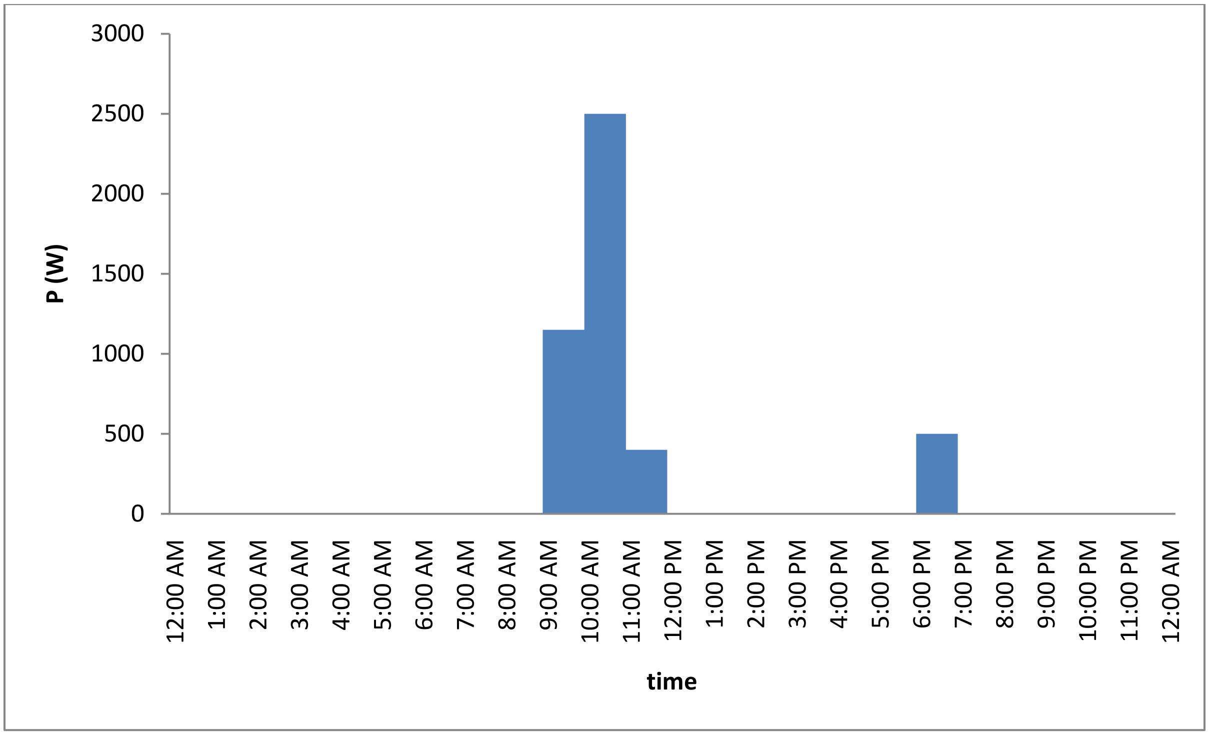


Figure 2.15. Daily electric power demand of an electric stove [23].

2.10.1 Operational Constraints

The stove's mathematical constraint equations are presented in [9], and are as follows:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = Stv \\ 0 & \text{if } t \notin T_i\,, i = Stv \end{cases} \qquad EOT_i \leq T_i \leq LOT_i \qquad (2.93)$$

$$U_i(t) - D_i(t) = S_i(t) - S_i(t-1) \qquad \forall t \in T_i\,, i = Stv \qquad (2.94)$$

$$U_i(t) - D_i(t) \leq 1 \qquad \forall t \in T_i\,, i = Stv \qquad (2.95)$$

$$\sum_{t \in T_i} S_i(k) = ROT_i \qquad \forall t \in T_i\,, i = Stv \qquad (2.96)$$

$$\sum_{k=t-1}^{t-MUT_i+1} U_i(k) \leq S_i(t) \qquad \forall t \in T_i\,, i = Stv \qquad (2.97)$$

These equations are simply trying to allocate the optimum stove working hours that would minimize the energy cost by moving the operating hours towards periods of low energy prices and are based on the following assumptions:

- The best stove operating hours are 8:00 AM to 12:00 PM.
- The stove is required to operate for a maximum of 4 hours per day.
- The stove should work for a minimum of two successive hours each time it turned on.

The above can differ from house to house according to the customer preferences; however, theses variations can be readily reflected in (2.93) to (2.97) by simply changing the appropriate limits. Moreover, it is well known that the resident's activity level has a direct effect on stove use. Therefore, activity level should also be included in the stove equations to improve this model. There are also more than one heating elements in the stove that might be running simultaneously during the cooking period; this situation could be included to produce a more advanced model.

2.10.2 Practical Implementation Issues

Household occupants use the stove any time they want; therefore, it is hard for utilities to manage stove time of use. For this reason, it was proposed in [9] to implement 2 colored pilot lamps to indicate the optimum stove time of use leaving the customer to take the final decision of using the stove. From the implementation point of view, this appliance should always be monitored by a two-way communication device to find out whether the customer has followed or disregarded the EHMS recommended ON/OFF decisions, as the EHMS optimization should be re-run each time there is a mismatch between the EHMS and customer decisions to re-calculate the new optimum decisions for other appliances.

2.11 Pool Pump

Information about the number of residential swimming pools and their energy consumption in Ontario are not available. Significant amount of energy is required for heating and maintaining water temperature in pools, in addition to the energy used by the pool pump to circulate and filter the pool water. Pool water heating can be a solar, gas or electrical heat pump. Residential swimming pool energy consumption in Australia, for example, is approximately 3.3% of total residential electricity use [24]. One can expect this energy consumption to be less in Ontario due to cold weather which leads to fewer numbers of swimming days per year, not to mention that pools water in Ontario is drained out after the swimming season. The break-up of electrical energy use in swimming pools is [24]:

- 76% for pumps,
- 6% for chlorination cells,
- 14% for electric heaters,
- 4% for timers and controls.

Generally 500 to 2000 W single phase pumps are used for residential swimming pools with 3 to 8 working hours per day for water filtration depending on pool size, pump size, environmental conditions such as outside temperature and sunlight illumination level, water filtration equipment, and how often the pool is used as well as the pool manufacturer recommendations. Usually, pool pumps are controlled by electro-mechanical or electronic ON/OFF clock timers with start- and end-time manually selected by users.

2.11.1 Operational Constraints

The pool pump mathematical constraint equations presented in [9] are as follows:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = Ppump \\ 0 & \text{if } t \notin T_i\,, i = Ppump \end{cases} \qquad EOT_i \ \leq \ T_i \ \leq \ LOT_i \tag{2.98}$$

$$U_i(t) - D_i(t) = S_i(t) - S_i(t-1) \qquad \forall t \in T_i\,, i = Ppump \tag{2.99}$$

$$U_i(t) - D_i(t) \leq 1 \qquad \forall t \in T_i\,, i = Ppump \tag{2.100}$$

$$\sum_{t \in T_i} S_i(k) = ROT_i \qquad \forall\, t \in T_i\,, i = Ppump \tag{2.101}$$

$$\sum_{k=t-1}^{t-MUT_i+1} U_i(k) \leq \ S_i(t) \qquad \forall\, t \in T_i\,, i = Ppump \tag{2.102}$$

$$\sum_{k=t-1}^{t+MDT_i+\mathbf{1}} D_i(k) \leq \ (1 - S_i(t)) \qquad \forall t \in T_i\,, i = Ppump \tag{2.103}$$

These equations try to allocate the optimum pump operating hours that would minimize the energy cost by moving these working hours towards off-peak periods based on the following assumptions:

- The pump should be allowed to work for at least 10 hours per day.
- The maximum numbers of successive ON states should be two hours.
- The minimum numbers of successive OFF states should be two hours.

This model could be improved as per the following considerations:

- The total number of pump working hours per day should be selected according to the pool size, pump size and pool manufacturer recommendations and other factors mentioned above.
- The maximum numbers of continuous successive ON states should not be restricted, as this constraint will deprive the solver from finding better possible solutions.
- The minimum number of continuous successive OFF states should not be restricted for the above reason.

Based on these considerations, only (2.98) and (2.101) equations are going to be implemented. Activity level could also be included in the pump equations to account for effect of residents' activity on pool swimming hours.

2.12 Lighting

Lighting loads are common in every house and accounted for 13% of total residential electricity use in Ontario in 2007 [3]. Lumen (lm) is the measuring unit of the power of light perceived by the human eye. A 23 W compact florescent lamp, for example, emits about 1500 lm [25]; this light thus has a conversion efficiency of 65 lm/W. The unit for measuring illumination is Lux (lx); one lx is equal to one lumen per square meter. If the 1500 lumens generated by the aforementioned 23 W light are concentrated on an area of 10 square meters, then that area will receive illumination of 150 lx. However, if the same 23W light is used to illuminate an area of 20 square meters, then the area will receive 75 lx illumination only.

Different house activities require different indoor illumination levels. For casual seeing, 100 lx is enough for a person of age 40 or less; whereas 200 lx is required past age 60 [26]. For reading and writing, 400 lx are required for ages 40 or less, while 800 lx is required past age 60 [26]. The required indoor illumination level can come from electric lighting sources plus the fraction of outdoor illumination that can enter the house through windows. Outdoor illumination levels change with the hour of the day, day of the year, house location and weather conditions. The outdoor light level is approximately 10,000 lx on a clear day; however, in a building in the area closest to the windows, the light level may be reduced to approximately 1,000 lx; in the middle area it may be as low as 25-50 lx [27]. The fraction of outdoor light illumination that can enter the house in a certain house zone depends on zone orientation, day of the year, size of the window and statues of the curtain condition whether open or closed.

2.12.1 Operational Constraints

The lightings constraint equations proposed in [9] are as follows:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i \,, i = LI \\ 0 & \text{if } t \notin T_i \,, i = LI \end{cases} \qquad EOT_i \leq T_i \leq LOT_i \tag{2.104}$$

$$IL(t) + IL_{out}(t) \geq (1 + K_t)\, IL_{req}(t) \qquad \forall t \in T_i \tag{2.105}$$

$$K_t = -0.2041\, C_t + 1.898 \qquad \forall t \in T_i \tag{2.106}$$

Where K_t is a coefficient that represents the dependence of additional lighting generated by electricity on the electricity price [9]. This variable is constrained in (2.106) to be equal to unity at off-peak electricity price of 4.4 cents/kWh and zero at on-peak electricity price of 9.3 cents/kWh. Constraint (2.105) ensures that the total illumination from the lighting system and outdoor daylight during off-peak periods is more than two times the required illumination, whereas the total illumination during on-peak periods is only equal to or more than the required illumination. It was assumed that the required illumination $IL_{req}(t)$, shown in Figure 2.16, represents the number of zones that are required to be illuminated in a typical house. $IL_{out}(t)$, shown in Figure 2.17, represents a normalized outdoor daylight illumination that can enter the house. The effect of the house occupancy on the lighting load is considered in the required illumination parameter. The objective is to minimize $IL(t)$, which represents the number of zones to be illuminated by electricity after considering the outdoor daylight illumination. The model assumes that the lighting load of the house can be divided into six zones.

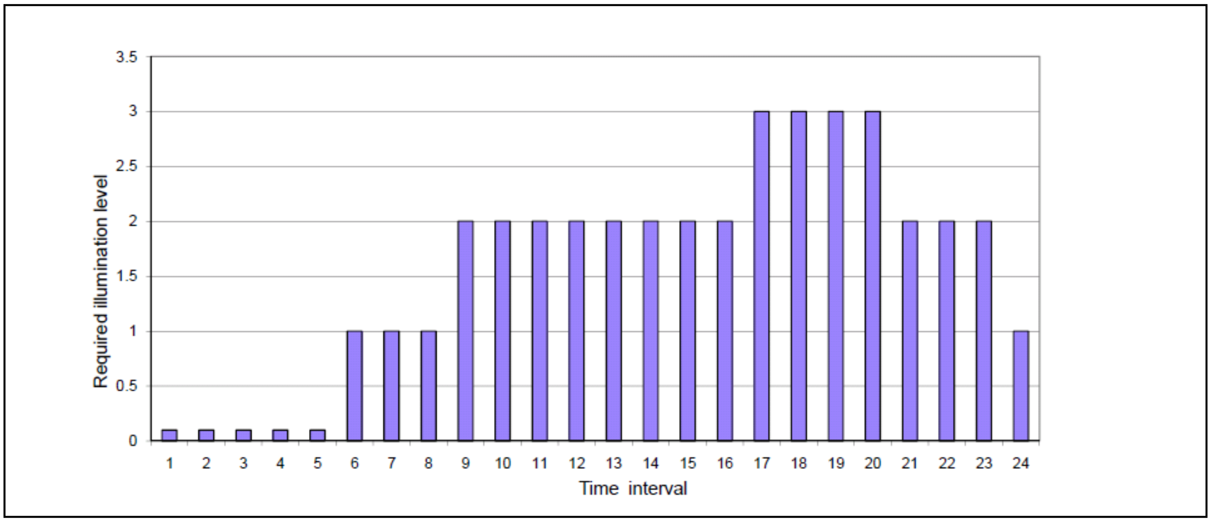


Figure 2.16: Example of a required illumination [9].

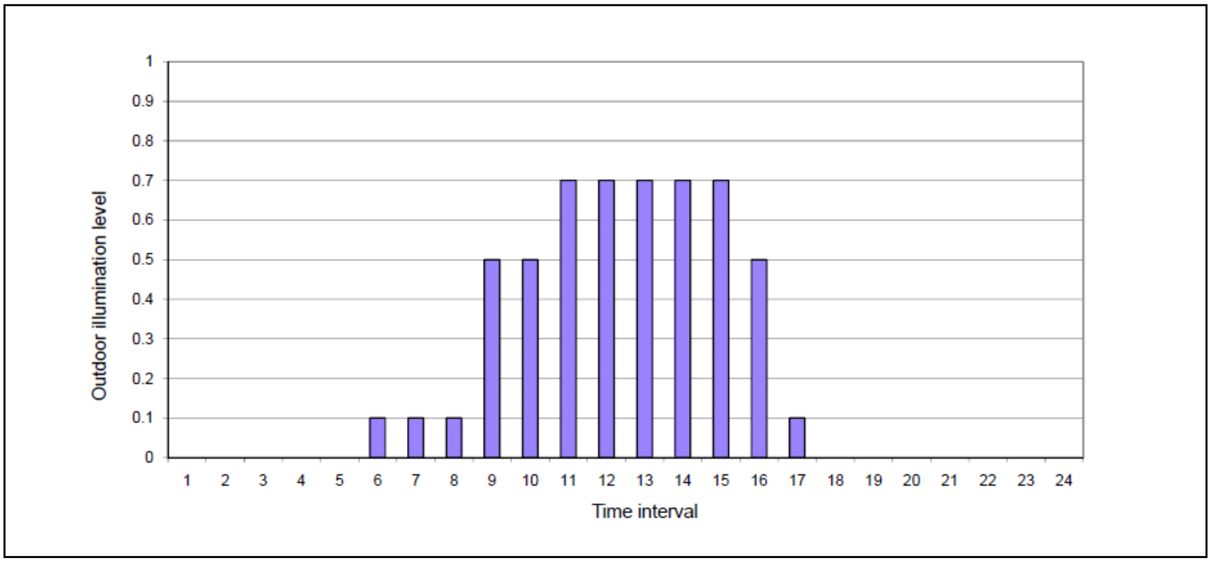


Figure 2.17. Example of outdoor illumination [9].

.

Model (2.104) to (2.106) produces an integer number that represents the number of zones to be illuminated by electricity without assigning these zones. An improvement to this model that could make the implementation easier can be achieved by dealing with each zone separately and declaring a binary variable to represent the ON/OFF decision of the zone as shown in the following set of equations:

$$S_i(t) = \begin{cases} 0 \text{ or } 1 & \text{if } t \in T_i\,, i = LI_j \\ 0 & \text{if } t \notin T_i\,, i = LI_j \end{cases} \qquad EOT_i \ \le\ T_i \ \le\ LOT_i \tag{2.107}$$

$$S_i(t) = \begin{cases} 0 \ \text{ if } \ IL_{out,j}(t) \ge & IL_{req,j}(t) \\ 1 & \text{otherwise} \end{cases} \qquad \forall t \in T_i\,, i = LI_j \tag{2.108}$$

$$IL_j(t) = \ S_i(t)\ IL_{req,j}(t) \qquad \forall t \in T_i\,, i = LI_j \tag{2.109}$$

where j is the zone number index, and $IL_{out,j}(t)$ represents the amount outdoor illumination entering the zone, which can be calculated or estimated for each zone in each house after a site visit. $IL_{erq_j}(t)$ is the required zone illumination level in lx. The optimum binary ON/OFF decisions $S_{LI_j}(t)$ generated by the EHMS can be transmitted to actuators to switch each zone ON or OFF individually to generate the additional illumination from electricity $IL_j(t)$.

A light dimmer could be used also instead of an actuator to generate the additional required illumination from electricity. In this case, the $S_{LI_j}(t)$ binary variables should be relaxed to free variables bounded between 0 and 1. Using actuators or dimmers for each zone separately may require some house electric wiring modifications which is expensive.

2.13 Summary

At the beginning of this chapter, the objective function of the optimization model was discussed in detail, followed by a comprehensive explanation and discussion of the concept of residential appliances activity level and the way of extracting the activity level information from the household hourly energy consumption. Then, the mathematical models of each appliance were discussed and the methods for calculating the different model parameters were explained. The following residential appliances were included in these discussions: water heater, refrigerator, cooling/heating systems, energy storage devices, stove, pool pump, dish washer, cloth washer and the dryer.

Chapter 3
Simulation and Experimental Results

3.1 GLPK Simulation Results

GLPK (GNU Linear Programming Kit) package was developed by Andrew Makhorin and is distributed as part of the GNU Project, under the GNU General Public License (GPL) [10]. GLPK is intended for solving large-scale LP problems, MILP problems, and other related problems. It is a set of routines written in ANSI C and organized in the form of a callable library [10]. GLPK supports the GNU MathProg modeling language (GMPL), which is a subset of the AMPL language. The GLPK package includes the following main components:

- primal and dual simplex methods,
- primal-dual interior-point method,
- branch-and-cut method,
- translator for GNU MathProg,
- application program interface (API),
- stand-alone LP/MIP solver.

GLPK contains nearly 100 callable routines for loading and modifying a problem instance, solving the loaded instance, querying the solver, and getting and setting algorithm parameters [10]. There are also utility routines to read and write files in MPS format, LP format, and GMPL. The user can set a parameter to choose from one of four branching methods: branching on first integer variable, branching on last integer variable, branching on most fractional variable, or branching using heuristics (which is the default method). The user can also change the search strategy to depth-first-search, breadth-first-search, best local bound search, or best projection heuristics (which is the default method).

GLPK version 4.38 [28] is used in the present work to solve the mathematical model of a sample house. The code CDMRes4, originally written in AMPL and solved using CPLEX in [9], is modified in this work into two versions: CDMRes4A and CDMR4B, written in GMPL and solved using GLPK. The modifications were based on the discussions presented in Chapter 2.

3.1.1 Code Version CDMRes4A.mod

In CDMRes4A.mod, which is shown in Appendix A, only minor modifications to the original code from [9] were performed as follows:

- All parameters in the data file are updated to more realistic values as shown in Table 3.1.
- The code calculates $HWU(t)$ using equation (2.25), and calculates the cooling/heating, dishwasher and refrigerator activity level $AL(t)$ using equations (2.2) and (2.40) assuming that the household hourly energy consumption data, $EEC(t)$, of the previous week are available.
- The minimum up/down time constraints on water heater, cooling/heating system, ESD and refrigerator are not considered.
- The washer equations are modified to include the effect of number of persons in the house on the washer storage level instead of their activity inside the house.
- The dryer equations are modified to include the effect of cloth washer operation on the dryer storage level instead of the resident's activity level as shown in (2.92).
- The non-linear constraints in the dishwasher and washer are replaced by the linear equations (2.74) and (2.85).

Table 3.1. Parameter settings in code CDMRes4A.mod.

Appliance	Parameter	Value	Calculation or measurement procedure
Refrigerator	P_{FR}	250 W	From name-plate data or by measurement
	θ_{FR}^{up}	8 ^{o}C	User defined
	θ_{FR}^{low}	2 ^{o}C	User defined
	EOT_{FR}	1	User defined
	LOT_{FR}	96	User defined
	β_{FR}	0.28	According to Section 2.4.2 (page 30)
	α_{FR}	5.5	According to Section 2.4.2 (page 29)
	γ_{FR}	1.21	According to Section 2.4.2 (page 29)

AC	P_{AC}	11500 W	From name-plate data or by measurement
	θ_{AC}^{up}	23 ^{o}C	User defined
	θ_{AC}^{low}	17 ^{o}C	User defined
	EOT_{AC}	1	User defined
	LOT_{AC}	96	User defined
	β_{AC}	0.0275	According to Section 2.5.2 (page 34)
	α_{AC}	0.51	According to Section 2.5.2 (page 34)
	γ_{AC}	0.0075	According to Section 2.5.2 (page 34)
Heating	P_H	400 W	According to gas heating system fan rating for gas space heating or heater name-plate data for electric heating system
	θ_{H}^{up}	Not applicable	θ_{AC}^{up} was used instead
	θ_{H}^{low}	Not applicable	θ_{AC}^{low} was used instead
	EOT_H	Not applicable	EOT_{AC} was used instead
	LOT_H	Not applicable	LOT_{AC} was used instead
	β_H	Not applicable	β_{AC} was used instead
	α_H	1.02	According to Section 2.5.2 (page 34)
	γ_H	Not applicable	γ_{AC} was used instead
Water Heater	P_{WH}	3600 W	From name-plate data
	θ_{WH}^{up}	65 ^{o}C	User defined
	θ_{WH}^{low}	55 ^{o}C	User defined
	EOT_{WH}	1	User defined
	LOT_{WH}	96	User defined
	β_{WH}	0.068	According to (2.24)
	α_{WH}	4.44	According to (2.18)
	γ_{WH}	0.083	According to (2.21) and (2.22)
Stove	P_{stv}	1500 W	From name-plate data
	EOT_{stv}	20	User defined
	LOT_{stv}	96	User defined
	ROT_{stv}	24	User defined
	MUT_{stv}	4	User defined
	$MSOT_{stv}$	16	User defined

Pool Pump	P_{Ppump}	750 W	From name-plate data
	EOT_{Ppump}	1	User defined
	LOT_{Ppump}	96	User defined
	ROT_{Ppump}	32	User defined
	MUT_{Ppump}	Not applicable	Not used
	MDT_{Ppump}	Not applicable	Not used
	$MSOT_{Ppump}$	Not applicable	Not used
ESD	P_{ESD}	100 W	From ESD name-plate data
	EOT_{ESD}	1	User defined
	LOT_{ESD}	96	User defined
	ESL_{ESD}	250 Wh	According to ESD manufacturer recommendation
	MUT_{ESD}	Not applicable	Not used
	MDT_{ESD}	Not applicable	Not used
Lighting	P_{IL}	150 W	From name-plate data
	EOT_{IL}	1	User defined
	LOT_{IL}	96	User defined
Dish Washer	P_{DW}	600 W	From name-plate data (average value per hour)
	EOT_{DW}	7	User defined
	LOT_{DW}	23	User defined
	MUT_{DW}	2	User defined
	MDT_{DW}	1	User defined
	β_{DW}	150	According to Section 2.7.2 (page 42)
	Cap_{DW}	100	Given (100%)
Washer	P_{W}	450 W	From name-plate data (average value per hour)
	EOT_{W}	7	User defined
	LOT_{W}	23	User defined
	MUT_{W}	2	User defined
	MDT_{W}	1	User defined
	β_{W}	Not applicable	Not used
	Cap_{W}	100	Given (100%)
	$person$	4	Number of persons living in a house

Dryer	P_D	1100 W	From name-plate data (average value per hour)
	EOT_D	7	User defined
	LOT_D	24	User defined
	MUT_D	2	User defined
	MDT_D	1	User defined
	$MTGap_D$	3	User defined
	Cap_D	100	Given (100%)

The following GLPK solver options were used as they were found useful in expediting the search for the optimum solution:

--intopt : enables MIP presolving.

--dfs : enables backtrack using depth first search.

--last : enables branch on last integer variable.

--gomory : generates Gomory's mixed integer cuts.

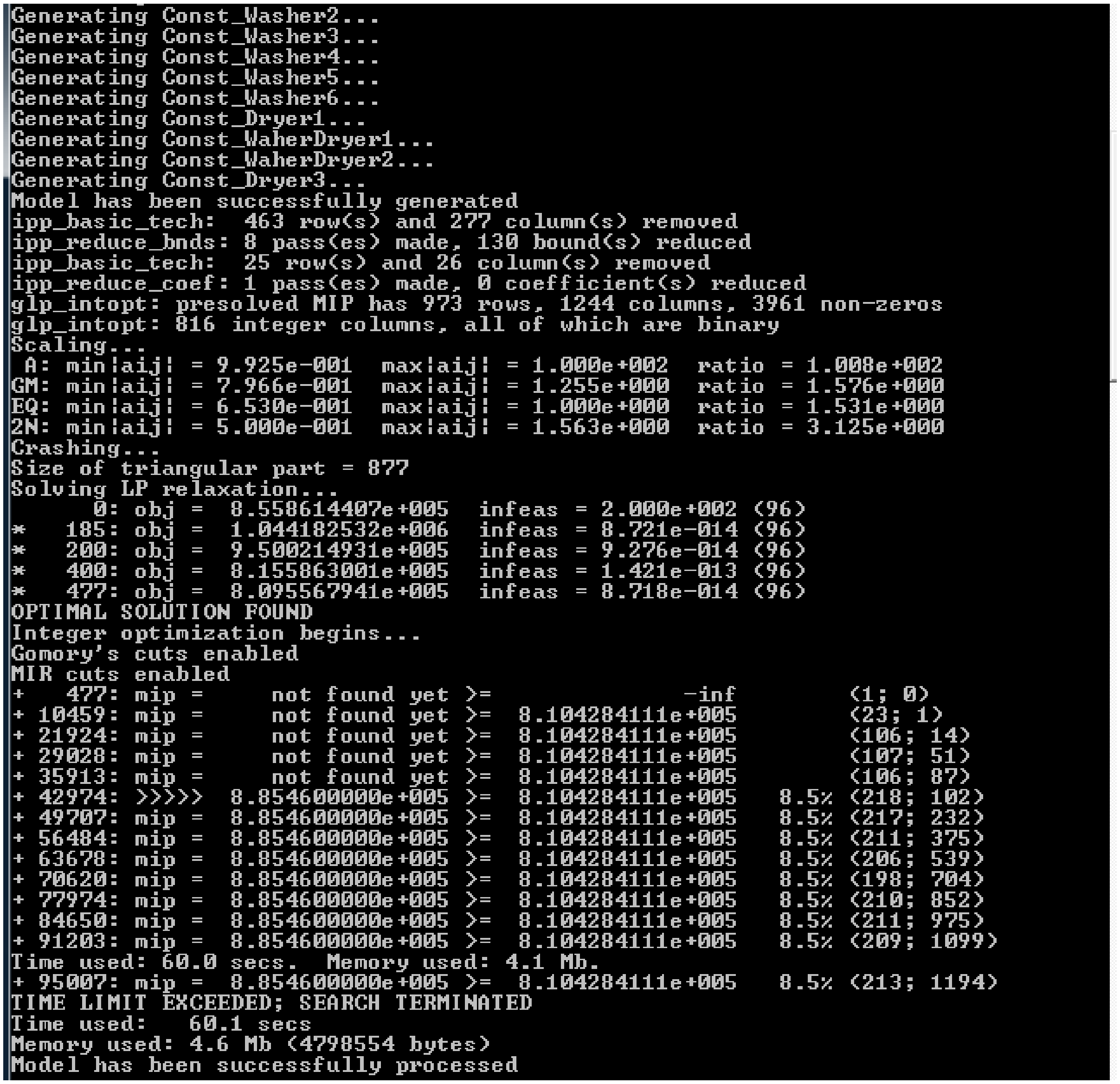
--mir : generates MIR (mixed integer rounding) cuts.

--tmlim 60: limits solution time to 60 seconds.

The GLPK solver terminal report is shown in Figure 3.1. The report has the following format:

```
+ nnn: mip  =  xxx   >=   yyy  gap  (ppp ;  qqq)
```

where "nnn" is the simplex iteration number; "xxx" is the objective function value for the best known integer feasible solution, "yyy" is a global bound for exact integer optimum and "gap" is the relative MIP gap in percents. The relative MIP "gap" is used to measure the quality of the best integer feasible solution found so far, "ppp" is the number of sub-problems in the active list and "qqq" is the number of sub-problems which have been already fathomed and therefore removed from the branch-and-bound search tree.



```
Generating Const_Washer2...
Generating Const_Washer3...
Generating Const_Washer4...
Generating Const_Washer5...
Generating Const_Washer6...
Generating Const_Dryer1...
Generating Const_WaherDryer1...
Generating Const_WaherDryer2...
Generating Const_Dryer3...
Model has been successfully generated
ipp_basic_tech:  463 row(s) and 277 column(s) removed
ipp_reduce_bnds: 8 pass(es) made, 130 bound(s) reduced
ipp_basic_tech:  25 row(s) and 26 column(s) removed
ipp_reduce_coef: 1 pass(es) made, 0 coefficient(s) reduced
glp_intopt: presolved MIP has 973 rows, 1244 columns, 3961 non-zeros
glp_intopt: 816 integer columns, all of which are binary
Scaling...
 A: min|aij| = 9.925e-001  max|aij| = 1.000e+002  ratio = 1.008e+002
GM: min|aij| = 7.966e-001  max|aij| = 1.255e+000  ratio = 1.576e+000
EQ: min|aij| = 6.530e-001  max|aij| = 1.000e+000  ratio = 1.531e+000
2N: min|aij| = 5.000e-001  max|aij| = 1.563e+000  ratio = 3.125e+000
Crashing...
Size of triangular part = 877
Solving LP relaxation...
      0: obj =  8.558614407e+005  infeas = 2.000e+002 (96)
*   185: obj =  1.044182532e+006  infeas = 8.721e-014 (96)
*   200: obj =  9.500214931e+005  infeas = 9.276e-014 (96)
*   400: obj =  8.155863001e+005  infeas = 1.421e-013 (96)
*   477: obj =  8.095567941e+005  infeas = 8.718e-014 (96)
OPTIMAL SOLUTION FOUND
Integer optimization begins...
Gomory's cuts enabled
MIR cuts enabled
+   477: mip =     not found yet >=              -inf        (1; 0)
+ 10459: mip =     not found yet >=  8.104284111e+005        (23; 1)
+ 21924: mip =     not found yet >=  8.104284111e+005        (106; 14)
+ 29028: mip =     not found yet >=  8.104284111e+005        (107; 51)
+ 35913: mip =     not found yet >=  8.104284111e+005        (106; 87)
+ 42974: >>>>>  8.854600000e+005 >=  8.104284111e+005   8.5% (218; 102)
+ 49707: mip =  8.854600000e+005 >=  8.104284111e+005   8.5% (217; 232)
+ 56484: mip =  8.854600000e+005 >=  8.104284111e+005   8.5% (211; 375)
+ 63678: mip =  8.854600000e+005 >=  8.104284111e+005   8.5% (206; 539)
+ 70620: mip =  8.854600000e+005 >=  8.104284111e+005   8.5% (198; 704)
+ 77974: mip =  8.854600000e+005 >=  8.104284111e+005   8.5% (210; 852)
+ 84650: mip =  8.854600000e+005 >=  8.104284111e+005   8.5% (211; 975)
+ 91203: mip =  8.854600000e+005 >=  8.104284111e+005   8.5% (209; 1099)
Time used: 60.0 secs.  Memory used: 4.1 Mb.
+ 95007: mip =  8.854600000e+005 >=  8.104284111e+005   8.5% (213; 1194)
TIME LIMIT EXCEEDED; SEARCH TERMINATED
Time used:   60.1 secs
Memory used: 4.6 Mb (4798554 bytes)
Model has been successfully processed
```

Figure 3.1. GLPK solver terminal report for code CDMRes4A.mod.

The first feasible solution was obtained after approximately 30 s, and the search for optimum solution was terminated after 60 s using the time limit option "--tmlim 60", because there was no progress in finding a better optimum solution as the relative gap settled at 8.5%. The solution was printed in a text file which was then exported to a spread sheet for analysis. Table 3.2 shows the resulting optimum decisions for the appliances which work on 96 15-minute intervals, whereas Table 3.3 shows the decisions for the appliance which work on 24 one-hour intervals.

Table 3.2. Appliances optimum ON/OFF decisions for CDMRes4A.mod.

Schedule	Time	AC	Heating	Water heat	Fridge	Stove	Pool pump	ESD	Lighting
1	12:00 AM	0	0	0	0	0	1	0	1
2	12:15 AM	0	0	0	0	0	0	0	1
3	12:30 AM	0	0	0	0	0	0	0	1
4	12:45 AM	0	0	0	1	0	0	0	1
5	1:00 AM	0	0	0	0	0	0	0	1
6	1:15 AM	0	0	0	0	0	0	0	1
7	1:30 AM	0	0	0	0	0	0	0	1
8	1:45 AM	0	0	0	1	0	0	0	1
9	2:00 AM	0	0	0	0	0	0	0	1
10	2:15 AM	0	0	0	0	0	1	0	1
11	2:30 AM	0	0	0	0	0	1	0	1
12	2:45 AM	0	0	0	1	0	1	0	1
13	3:00 AM	0	0	0	0	0	1	0	1
14	3:15 AM	0	1	1	0	0	1	0	1
15	3:30 AM	0	0	0	0	0	1	0	1
16	3:45 AM	0	0	0	0	0	1	0	1
17	4:00 AM	0	0	0	1	0	1	0	1
18	4:15 AM	0	1	0	0	0	1	0	1
19	4:30 AM	0	0	0	0	0	1	0	1
20	4:45 AM	0	0	0	0	1	1	0	1
21	5:00 AM	0	1	0	0	1	1	0	2
22	5:15 AM	0	1	1	1	0	1	0	2
23	5:30 AM	0	1	0	0	1	1	0	2
24	5:45 AM	0	1	0	0	1	1	0	2
25	6:00 AM	0	1	1	0	1	1	0	2
26	6:15 AM	0	1	0	1	1	1	0	2
27	6:30 AM	0	1	1	0	1	1	0	2
28	6:45 AM	0	0	0	0	1	1	0	2
29	7:00 AM	0	1	1	0	0	0	1	1
30	7:15 AM	0	0	0	0	0	0	1	1
31	7:30 AM	0	0	1	1	0	0	1	1
32	7:45 AM	0	0	0	0	0	0	1	1
33	8:00 AM	0	0	1	0	0	0	1	2
34	8:15 AM	0	0	0	1	0	0	1	2
35	8:30 AM	0	0	0	0	0	0	1	2
36	8:45 AM	0	0	0	0	0	0	1	2
37	9:00 AM	0	0	1	0	0	0	1	2
38	9:15 AM	0	0	0	1	0	0	1	2
39	9:30 AM	0	0	0	0	0	0	1	2
40	9:45 AM	0	0	0	0	0	0	1	2
41	10:00 AM	0	0	0	0	0	0	1	1
42	10:15 AM	0	0	0	1	0	0	1	1
43	10:30 AM	0	0	0	0	0	0	1	1
44	10:45 AM	0	0	0	0	0	0	1	1
45	11:00 AM	0	0	1	1	0	0	0	2
46	11:15 AM	0	0	0	0	0	0	0	2
47	11:30 AM	0	0	0	1	0	0	0	2
48	11:45 AM	0	0	0	0	0	0	0	2

49	12:00 PM	0	0	0	1	0	0	0	2
50	12:15 PM	0	0	0	1	0	0	0	2
51	12:30 PM	0	0	1	0	0	0	0	2
52	12:45 PM	0	0	0	1	0	0	0	2
53	1:00 PM	0	0	0	0	0	0	0	2
54	1:15 PM	0	0	0	0	0	0	1	2
55	1:30 PM	0	0	1	1	0	0	1	2
56	1:45 PM	0	1	0	0	0	0	1	2
57	2:00 PM	0	1	0	0	0	0	1	2
58	2:15 PM	0	0	0	0	0	0	1	2
59	2:30 PM	0	0	0	0	0	0	1	2
60	2:45 PM	0	0	0	1	0	0	1	2
61	3:00 PM	0	0	1	0	0	0	1	3
62	3:15 PM	0	1	1	0	0	0	1	3
63	3:30 PM	0	1	0	0	0	0	1	3
64	3:45 PM	0	0	0	1	0	0	1	3
65	4:00 PM	0	0	0	0	1	0	1	4
66	4:15 PM	0	0	0	0	1	0	1	4
67	4:30 PM	0	1	1	0	1	0	1	4
68	4:45 PM	0	1	1	1	1	0	1	4
69	5:00 PM	0	0	0	0	0	0	1	3
70	5:15 PM	0	0	0	0	0	0	1	3
71	5:30 PM	0	0	0	0	0	0	1	3
72	5:45 PM	0	0	1	0	0	0	1	3
73	6:00 PM	0	0	0	1	0	0	1	3
74	6:15 PM	0	0	0	0	0	0	1	3
75	6:30 PM	0	0	1	1	0	0	1	3
76	6:45 PM	0	0	0	0	0	0	1	3
77	7:00 PM	0	0	0	0	0	0	1	3
78	7:15 PM	0	0	0	1	0	0	1	3
79	7:30 PM	0	0	0	0	0	0	1	2
80	7:45 PM	0	0	0	0	0	0	1	2
81	8:00 PM	0	0	1	0	0	0	1	2
82	8:15 PM	0	0	0	1	0	0	1	2
83	8:30 PM	0	0	0	0	0	0	1	2
84	8:45 PM	0	1	0	0	0	0	1	2
85	9:00 PM	0	1	1	0	1	1	0	4
86	9:15 PM	0	0	0	1	1	1	1	4
87	9:30 PM	0	0	0	0	1	1	1	4
88	9:45 PM	0	0	0	0	1	1	0	4
89	10:00 PM	0	0	0	0	1	1	0	4
90	10:15 PM	0	0	0	1	1	1	0	4
91	10:30 PM	0	1	0	0	1	1	0	4
92	10:45 PM	0	0	0	0	1	1	1	4
93	11:00 PM	0	0	0	0	1	1	1	2
94	11:15 PM	0	0	0	0	1	1	1	2
95	11:30 PM	0	0	0	1	1	1	1	2
96	11:45 PM	0	1	1	0	1	1	1	2

Table 3.3. GLPK optimum decisions of dishwasher, washer and dryer.

Hour	Dish washer decisions	Dish washer storage level (%)	Cloth washer decisions	Cloth washer storage level (%)	Dryer decisions	Dryer storage level (%)
1	0	10	0	85	0	0
2	0	10	0	86	0	0
3	0	10	0	87	0	0
4	0	10	0	88	0	0
5	0	10	0	89	0	0
6	0	10	0	90	0	0
7	0	10	0	91	0	0
8	0	17	0	92	0	0
9	0	19	0	93	0	0
10	0	22	0	94	0	0
11	0	34	0	95	0	0
12	0	66	0	96	0	0
13	0	113	0	97	0	0
14	0	115	0	98	0	0
15	0	115	0	99	0	0
16	0	115	0	100	0	0
17	0	120	0	101	0	0
18	1	123	0	102	0	0
19	1	95	1	103	0	0
20	0	48	1	54	0	50
21	0	52	0	5	1	100
22	0	59	0	6	1	50
23	0	59	0	7	0	0
24	0	59	0	8	0	0

It was assumed in this analysis that the outdoor temperature is for the case of an Ontarian winter day as shown in Figure 3.2; for this reason one can see in Table 3.2 that only the heating system is working while the cooling system is idle. The ON decisions and the temperature with its maximum and minimum limits of the refrigerator, heating system and water heater are shown in Figure 3.3, Figure 3.4 and Figure 3.5 respectively. The dishwasher, washer and dryer storage levels are shown in Figure 3.6 and Figure 3.7, whereas the ESD storage levels and discharging decisions are shown in Figure 3.8. The lighting systems illumination levels are depicted in Figure 3.9. The total household electrical energy consumption is shown in Figure 3.10, where it can be clearly observed how the electrical energy consumption is shifted to periods of cheaper energy prices.

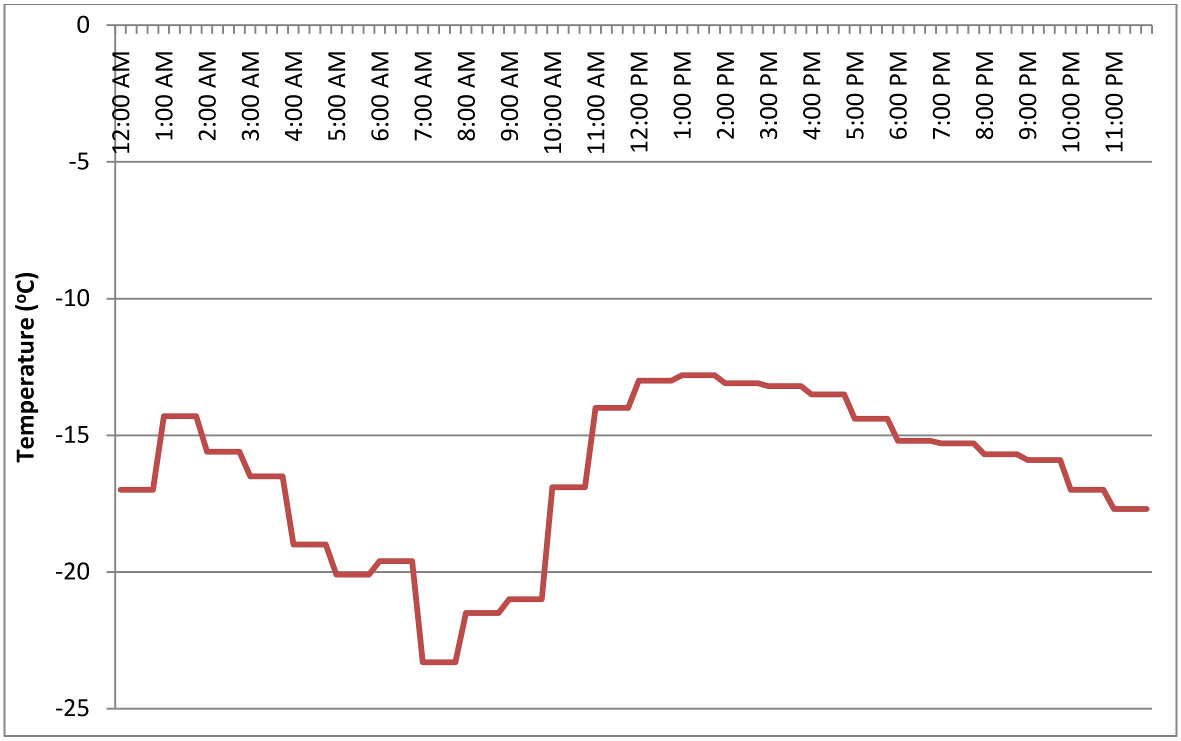


Figure 3.2. Outdoor temperature used in the analysis.

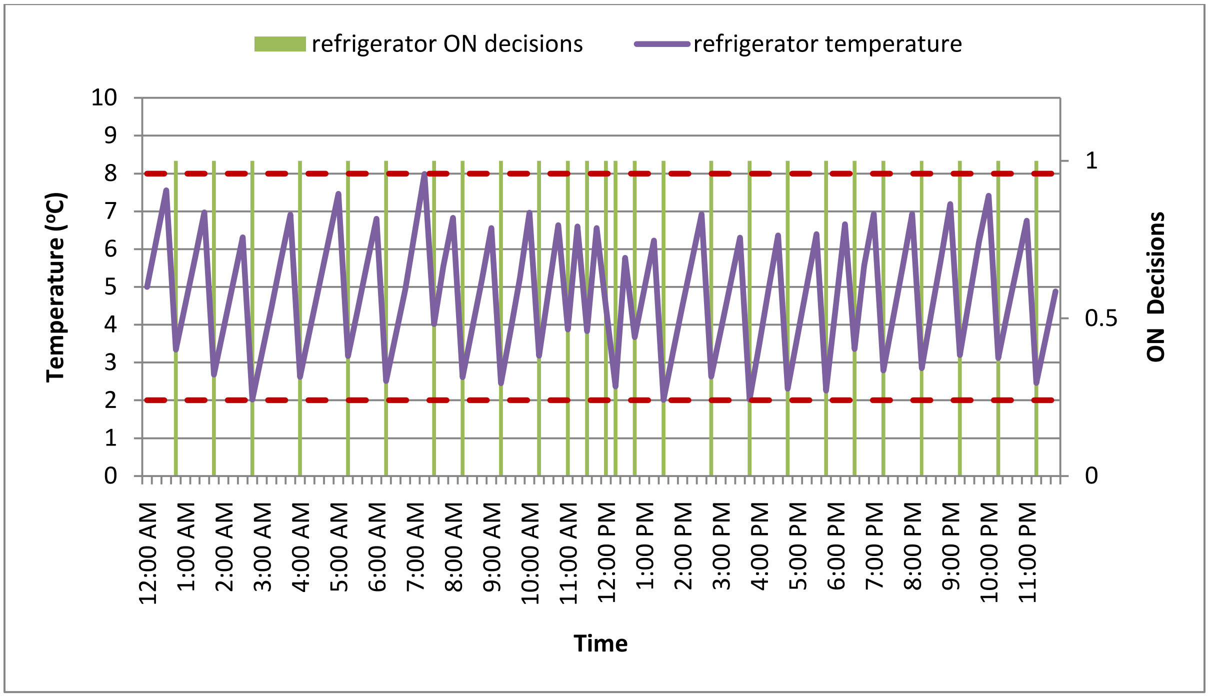


Figure 3.3. Refrigerator temperature and ON decisions.

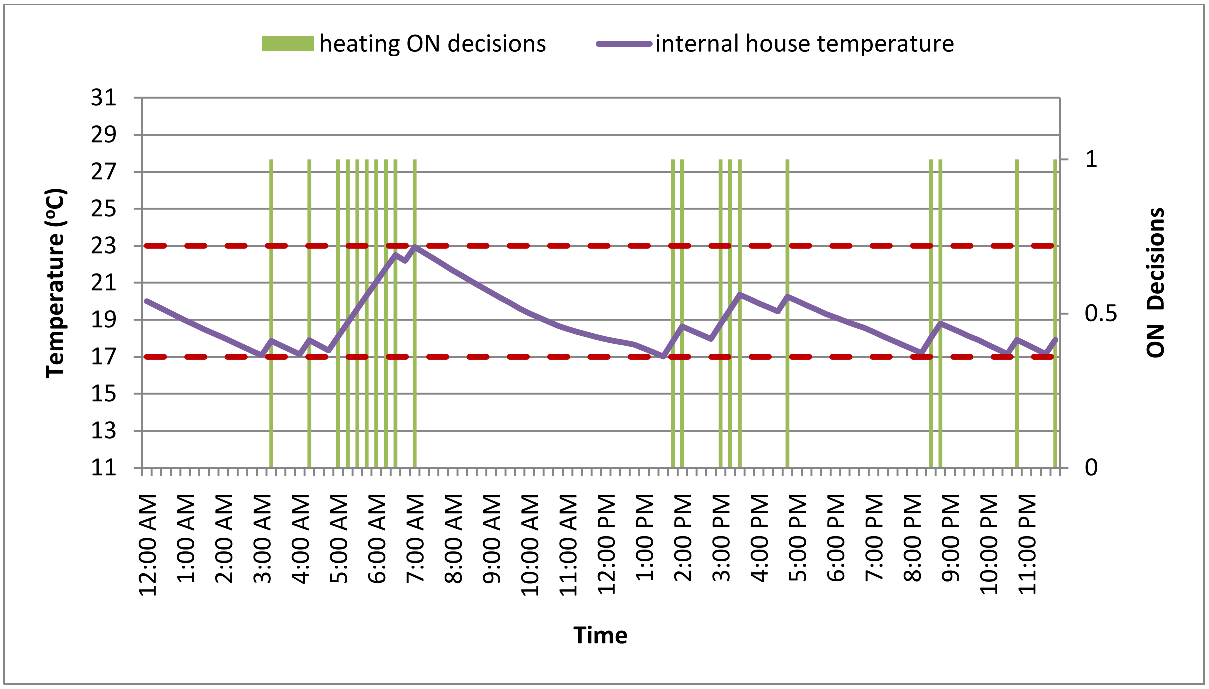


Figure 3.4. Heating system temperature and ON decisions.

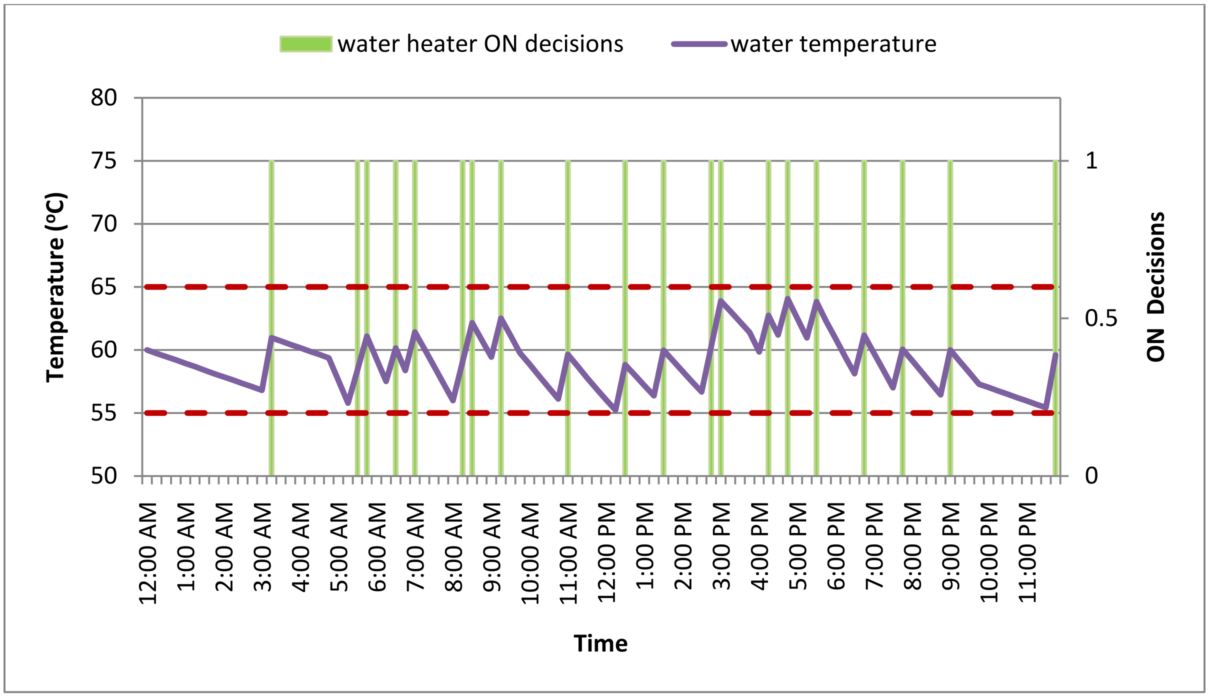


Figure 3.5. Water heater temperature and ON decisions.

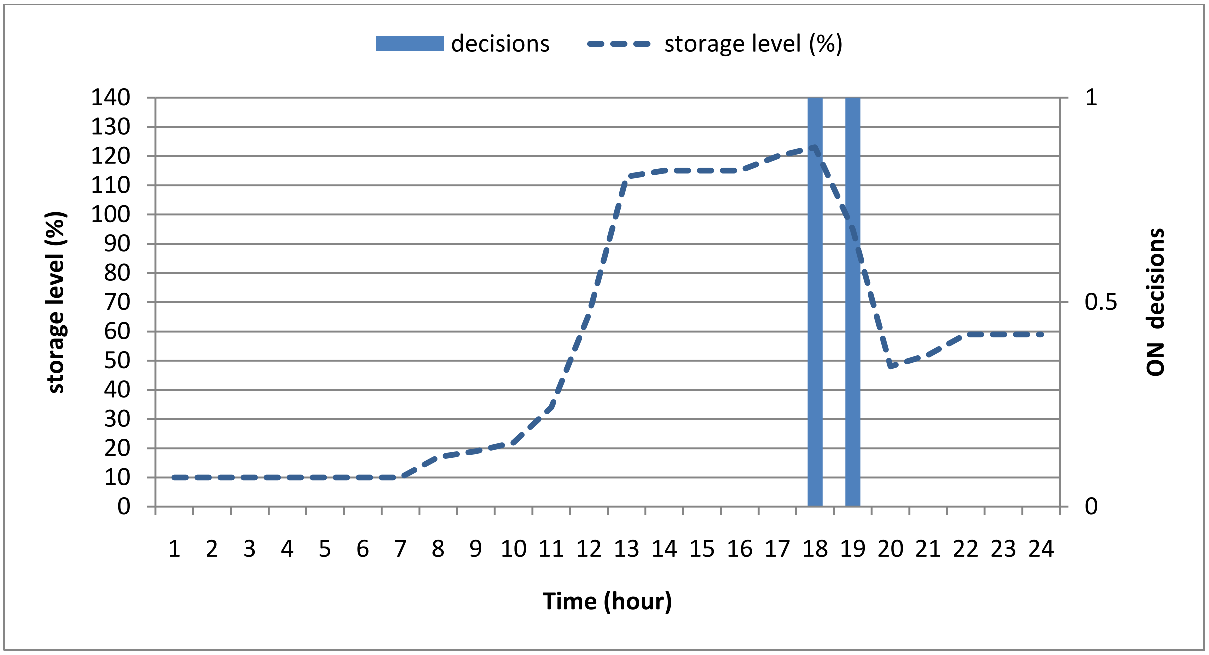


Figure 3.6. Dishwasher storage level and ON decisions.

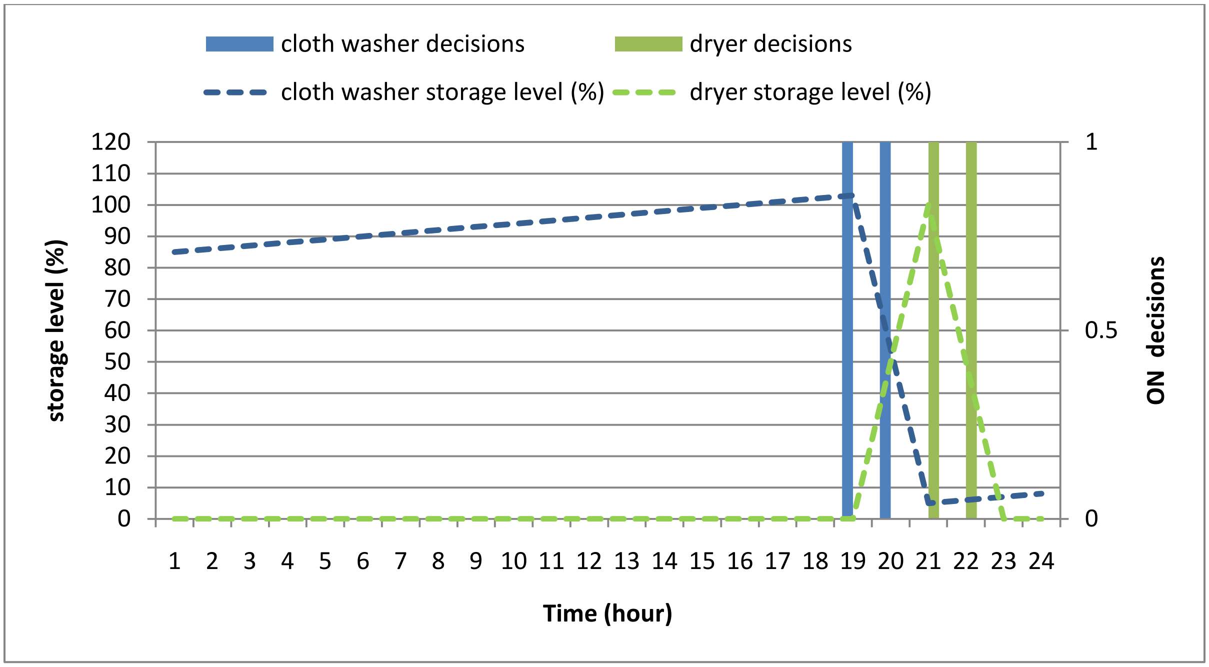


Figure 3.7. Cloth washer and dryer storage levels and ON decisions.

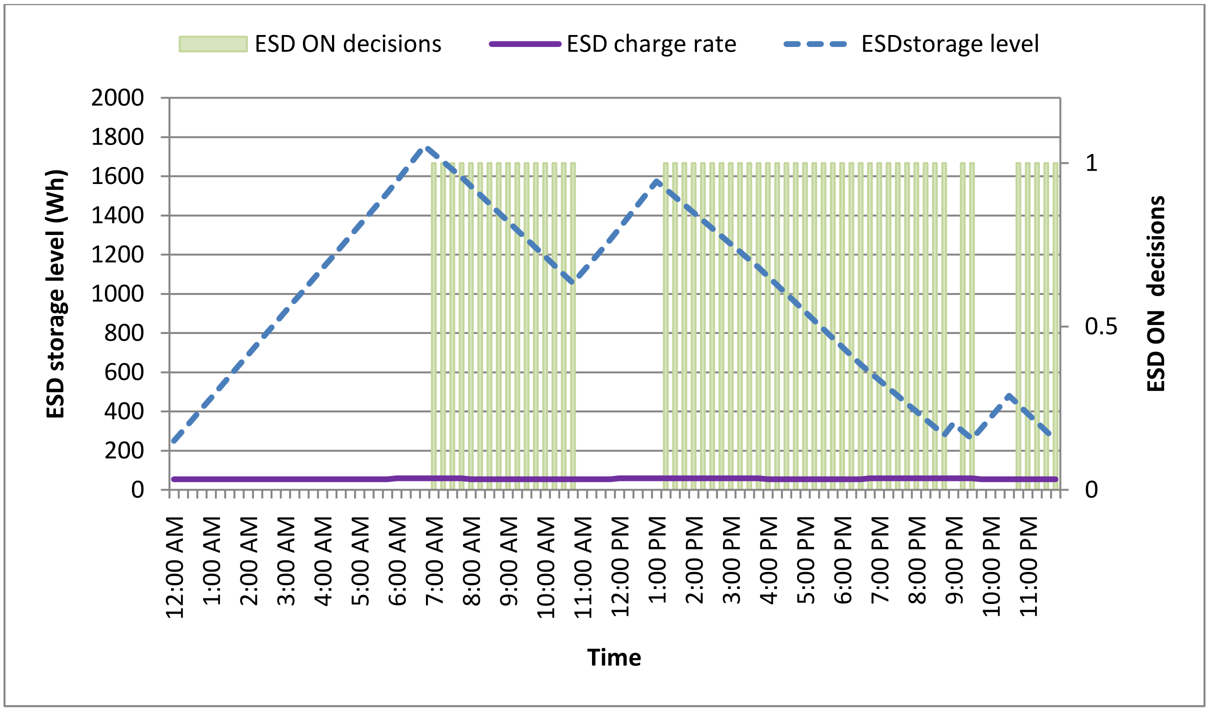


Figure 3.8. ESD discharging decisions and storage level.

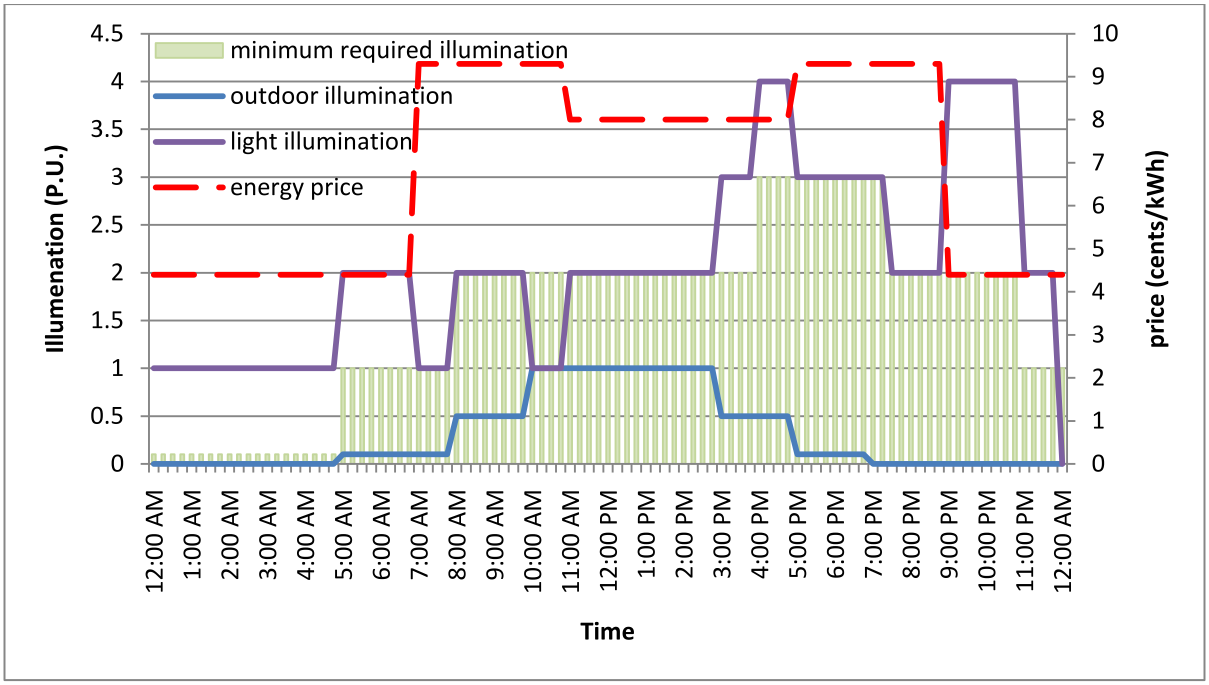


Figure 3.9. Lighting illumination.

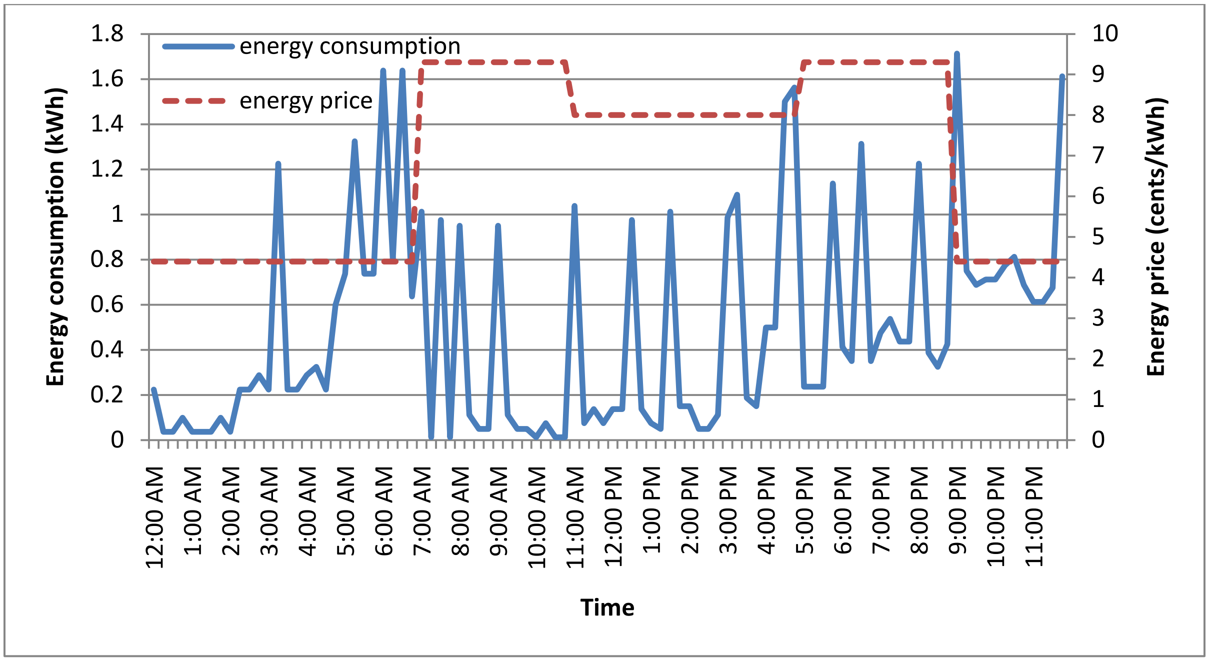


Figure 3.10. Total household electrical energy consumption.

3.1.2 Code Version CDMRes4B.mod

In code version, CDMRes4B.mod, which is given in Appendix B, the following additional modifications are performed to the previous CDMRes4A.mod code:

- The lighting model has been segregated into six standalone zones.
- The dishwasher, cloth washer and dryer equations are simplified so that they can be allowed to run during a maximum of 2 successive hours each day without relating these decisions to the storage level or activity level as discussed in Sections 2.7.2, 2.8.2 and 2.9.1.
- A complete domestic PV model was used instead of just an ESD device.
- A total of 96 time intervals were used for the dishwasher, washer and dryer, similar to all other appliances.

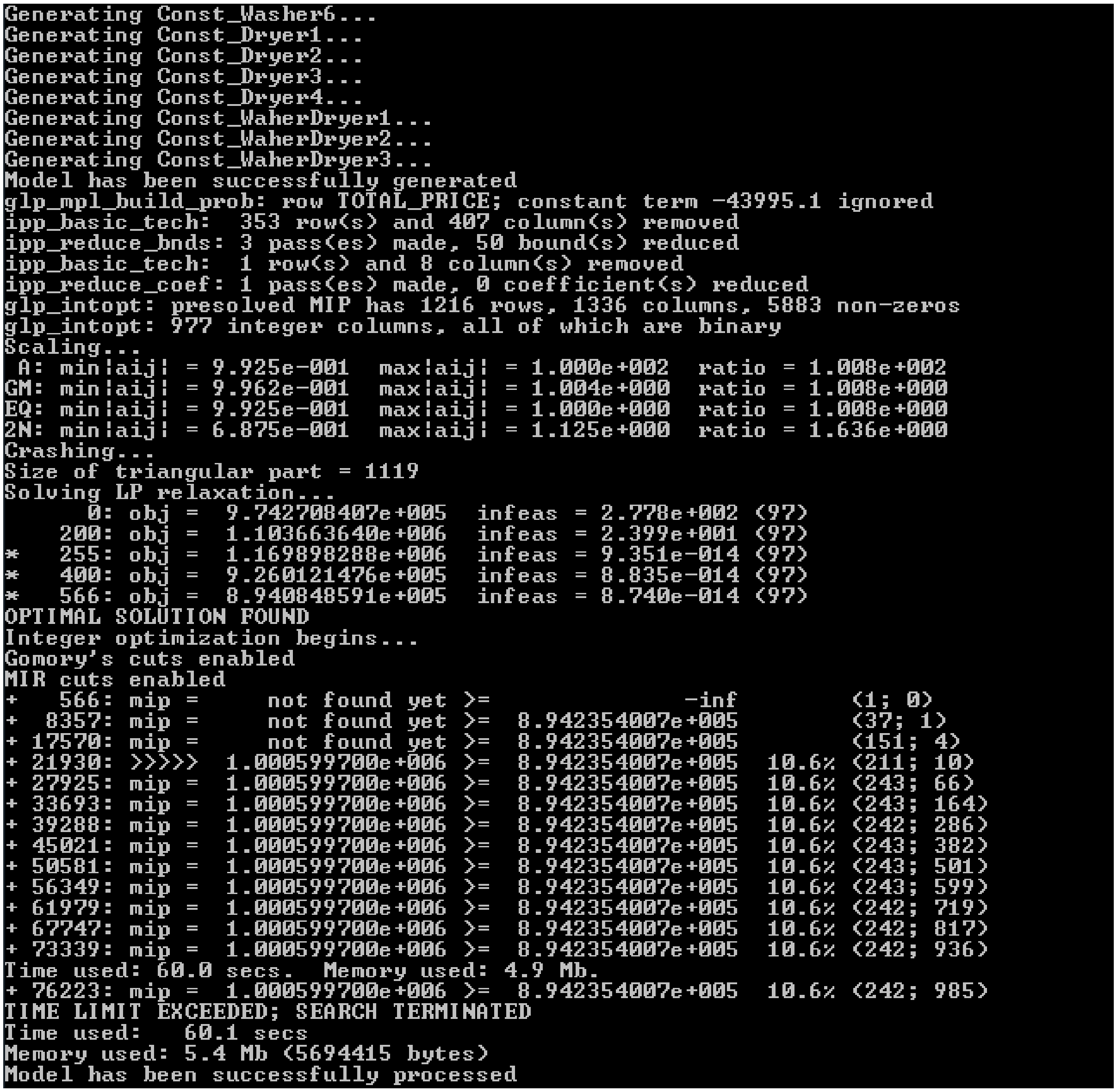
Table 3.4. Parameter settings in code CDMRes4B.mod.

Appliance	Parameter	Value	Calculation or measurement procedure
Refrigerator	P_{FR}	250 W	From name-plate data or by measurement
	θ_{FR}^{up}	8 ^{o}C	User defined
	θ_{FR}^{low}	2 ^{o}C	User defined
	EOT_{FR}	1	User defined
	LOT_{FR}	96	User defined
	β_{FR}	0.28	According to Section 2.4.2 (page 30)
	α_{FR}	5.5	According to Section 2.4.2 (page 29)
	γ_{FR}	1.21	According to Section 2.4.2 (page 29)
AC	P_{AC}	11500 W	From name-plate data or by measurement
	θ_{AC}^{up}	23 ^{o}C	User defined
	θ_{AC}^{low}	17 ^{o}C	User defined
	EOT_{AC}	1	User defined
	LOT_{AC}	96	User defined
	β_{AC}	0.0275	According to Section 2.5.2 (page 34)
	α_{AC}	0.51	According to Section 2.5.2 (page 34)
	γ_{AC}	0.0075	According to Section 2.5.2 (page 34)

Heating	P_H	400 W	According to gas heating system fan rating for gas space heating or heater name-plate data for electric heating system
	α_H	1.02	According to Section 2.5.2 (page 34)
Water Heater	P_{WH}	3600 W	From name-plate data
	θ_{WH}^{up}	65 ^{o}C	User defined
	θ_{WH}^{low}	55 ^{o}C	User defined
	EOT_{WH}	1	User defined
	LOT_{WH}	96	User defined
	β_{WH}	0.068	According to (2.24)
	α_{WH}	4.44	According to (2.18)
	γ_{WH}	0.083	According to (2.21) and (2.22)
Stove	P_{stv}	1500 W	From name-plate data
	EOT_{stv}	20	User defined
	LOT_{stv}	96	User defined
	ROT_{stv}	24	User defined
	MUT_{stv}	4	User defined
	$MSOT_{stv}$	16	User defined
Pool Pump	P_{Ppump}	750 W	From name-plate data
	EOT_{Ppump}	1	User defined
	LOT_{Ppump}	96	User defined
	ROT_{Ppump}	32	User defined
PV	P_{DR}	100 W	According to DC/AC inverter rating
	P_{CR}	55 W	According to battery charger rating
	EOT_{PV}	1	User defined
	LOT_{PV}	96	User defined
	$BATESL^{min}$	250 Wh	According to manufacturer recommendation
	$BATESL^{max}$	1500 Wh	According to battery capacity rating
Lighting1 to Lighting6	P_{IL1}	1-0.3 W/lx	Zone light conversion efficiency (measured)
	EOT_{IL}	1	User defined
	LOT_{IL}	96	User defined

Dish Washer	P_{DW}	600 W	From name-plate data (average value per hour)
	EOT_{DW}	64	User defined
	LOT_{DW}	92	User defined
	MUT_{DW}	8	User defined
	MDT_{DW}	4	User defined
	$MSOT_{DW}$	8	User defined
Washer	P_W	450 W	From name-plate data (average value per hour)
	EOT_W	64	User defined
	LOT_W	92	User defined
	MUT_W	8	User defined
	MDT_W	4	User defined
	$MSOT_W$	8	User defined
Dryer	P_D	1100 W	From name-plate data (average value per hour)
	EOT_D	64	User defined
	LOT_D	92	User defined
	MUT_D	8	User defined
	MDT_D	4	User defined
	$MTGap_D$	12	User defined

The solver terminal report shows that the first MILP feasible solution is obtained after approximately 15 s, which is faster than the previous code. The search for the optimum solution was terminated after 60 s using the option “--tmlim 60” as the relative gap settled at 10.6%. Figure 3.11 shows the solver terminal report and Table 3.5 shows resulting optimum decisions for all appliances. The relevant variables of the refrigerator, heating system and water heater are depicted in Figure 3.12,Figure 3.13 and Figure 3.14 respectively, while the dishwasher, cloth washer and dryer decisions are shown in Figure 3.15 and Figure 3.16. Household zones illuminations are illustrated in Figure 3.18 to Figure 3.23. The results for the domestic PV are shown in Figure 3.17, where one can see how the optimizer is trying to charge the battery during off-peak period, as much as its capacity allows, and then discharges the energy stored inside the battery into the grid during on-peak periods in order to maximize the cost savings. Finally the total household electrical energy consumption is shown in Figure 3.24, where it can be clearly noticed that the consumption has been shifted from on-peak periods to off-peak periods.



```
Generating Const_Washer6...
Generating Const_Dryer1...
Generating Const_Dryer2...
Generating Const_Dryer3...
Generating Const_Dryer4...
Generating Const_WaherDryer1...
Generating Const_WaherDryer2...
Generating Const_WaherDryer3...
Model has been successfully generated
glp_mpl_build_prob: row TOTAL_PRICE; constant term -43995.1 ignored
ipp_basic_tech:  353 row(s) and 407 column(s) removed
ipp_reduce_bnds: 3 pass(es) made, 50 bound(s) reduced
ipp_basic_tech:  1 row(s) and 8 column(s) removed
ipp_reduce_coef: 1 pass(es) made, 0 coefficient(s) reduced
glp_intopt: presolved MIP has 1216 rows, 1336 columns, 5883 non-zeros
glp_intopt: 977 integer columns, all of which are binary
Scaling...
 A: min|aij| = 9.925e-001  max|aij| = 1.000e+002  ratio = 1.008e+002
GM: min|aij| = 9.962e-001  max|aij| = 1.004e+000  ratio = 1.008e+000
EQ: min|aij| = 9.925e-001  max|aij| = 1.000e+000  ratio = 1.008e+000
2N: min|aij| = 6.875e-001  max|aij| = 1.125e+000  ratio = 1.636e+000
Crashing...
Size of triangular part = 1119
Solving LP relaxation...
      0: obj =  9.742708407e+005  infeas = 2.778e+002 (97)
    200: obj =  1.103663640e+006  infeas = 2.399e+001 (97)
*   255: obj =  1.169898288e+006  infeas = 9.351e-014 (97)
*   400: obj =  9.260121476e+005  infeas = 8.835e-014 (97)
*   566: obj =  8.940848591e+005  infeas = 8.740e-014 (97)
OPTIMAL SOLUTION FOUND
Integer optimization begins...
Gomory's cuts enabled
MIR cuts enabled
+   566: mip =     not found yet >=              -inf        (1; 0)
+  8357: mip =     not found yet >=  8.942354007e+005        (37; 1)
+ 17570: mip =     not found yet >=  8.942354007e+005        (151; 4)
+ 21930: >>>>>  1.000599700e+006 >=  8.942354007e+005  10.6% (211; 10)
+ 27925: mip =  1.000599700e+006 >=  8.942354007e+005  10.6% (243; 66)
+ 33693: mip =  1.000599700e+006 >=  8.942354007e+005  10.6% (243; 164)
+ 39288: mip =  1.000599700e+006 >=  8.942354007e+005  10.6% (242; 286)
+ 45021: mip =  1.000599700e+006 >=  8.942354007e+005  10.6% (243; 382)
+ 50581: mip =  1.000599700e+006 >=  8.942354007e+005  10.6% (243; 501)
+ 56349: mip =  1.000599700e+006 >=  8.942354007e+005  10.6% (243; 599)
+ 61979: mip =  1.000599700e+006 >=  8.942354007e+005  10.6% (242; 719)
+ 67747: mip =  1.000599700e+006 >=  8.942354007e+005  10.6% (242; 817)
+ 73339: mip =  1.000599700e+006 >=  8.942354007e+005  10.6% (242; 936)
Time used: 60.0 secs.  Memory used: 4.9 Mb.
+ 76223: mip =  1.000599700e+006 >=  8.942354007e+005  10.6% (242; 985)
TIME LIMIT EXCEEDED; SEARCH TERMINATED
Time used:   60.1 secs
Memory used: 5.4 Mb (5694415 bytes)
Model has been successfully processed
```

Figure 3.11. Solver terminal report of code CDMRes4B.mod.

Table 3.5. Appliances optimum ON/OFF decisions for CDMRes4B.mod.

Schedule	Time	AC	Heating	Water heater	Fridge	Stove	pool pump	Dish washer	Washer	Dryer	Light 1	Light 2	Light 3	Light 4	Light 5	Light 6	DR (discharge)	CR (charge)
1	12:00	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
2	12:15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	12:30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	12:45	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
5	1:00 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1:15 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	1:30 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	1:45 AM	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
9	2:00 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	2:15 AM	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
11	2:30 AM	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
12	2:45 AM	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0
13	3:00 AM	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
14	3:15 AM	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	0	0
15	3:30 AM	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
16	3:45 AM	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
17	4:00 AM	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0
18	4:15 AM	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
19	4:30 AM	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
20	4:45 AM	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0
21	5:00 AM	0	1	0	0	1	1	0	0	0	1	0	0	1	1	0	0	0
22	5:15 AM	0	1	0	1	1	1	0	0	0	1	0	1	1	1	0	0	0
23	5:30 AM	0	1	1	0	1	1	0	0	0	1	1	0	1	1	0	0	0
24	5:45 AM	0	1	1	0	0	1	0	0	0	1	1	0	0	1	0	0	0
25	6:00 AM	0	1	0	0	1	1	0	0	0	1	0	0	1	1	0	0	0
26	6:15 AM	0	1	0	1	1	1	0	0	0	1	0	1	1	1	0	0	1
27	6:30 AM	0	1	1	0	1	1	0	0	0	1	1	0	1	1	0	0	1
28	6:45 AM	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	1
29	7:00 AM	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0
30	7:15 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
31	7:30 AM	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
32	7:45 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
33	8:00 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
34	8:15 AM	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	1
35	8:30 AM	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
36	8:45 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37	9:00 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
38	9:15 AM	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0
39	9:30 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
40	9:45 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
41	10:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
42	10:15	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
43	10:30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
44	10:45	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

45	11:00 AM	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	1
46	11:15 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
47	11:30 AM	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1
48	11:45 AM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
49	12:00 PM	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1
50	12:15 PM	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1
51	12:30 PM	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
52	12:45 PM	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1
53	1:00 PM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
54	1:15 PM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
55	1:30 PM	0	0	1	1	1	0	0	0	0	0	1	1	1	0	0	0	1
56	1:45 PM	0	1	0	0	1	0	0	0	0	1	0	0	1	0	0	0	1
57	2:00 PM	0	1	0	0	1	0	0	0	0	1	0	0	1	0	0	0	1
58	2:15 PM	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1
59	2:30 PM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
60	2:45 PM	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	1
61	3:00 PM	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	1
62	3:15 PM	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
63	3:30 PM	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
64	3:45 PM	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	1
65	4:00 PM	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1
66	4:15 PM	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1
67	4:30 PM	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
68	4:45 PM	0	1	1	1	0	0	0	1	0	1	1	1	0	0	0	0	1
69	5:00 PM	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
70	5:15 PM	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
71	5:30 PM	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0
72	5:45 PM	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	1	0
73	6:00 PM	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
74	6:15 PM	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
75	6:30 PM	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	1	1
76	6:45 PM	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	1	0
77	7:00 PM	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
78	7:15 PM	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	1	0
79	7:30 PM	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
80	7:45 PM	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0
81	8:00 PM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
82	8:15 PM	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0
83	8:30 PM	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0
84	8:45 PM	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0
85	9:00 PM	0	0	1	0	1	1	1	0	0	0	1	0	1	1	1	0	0
86	9:15 PM	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
87	9:30 PM	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	0	0
88	9:45 PM	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	0	0
89	10:00 PM	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	0	0
90	10:15 PM	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
91	10:30 PM	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	0	0
92	10:45 PM	0	1	0	0	1	1	1	0	0	1	0	0	1	1	1	0	0
93	11:00 PM	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0
94	11:15 PM	0	0	0	1	1	1	0	0	0	0	0	1	1	1	0	0	0
95	11:30 PM	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0
96	11:45 PM	0	1	1	0	1	1	0	0	0	1	1	0	1	1	0	0	0

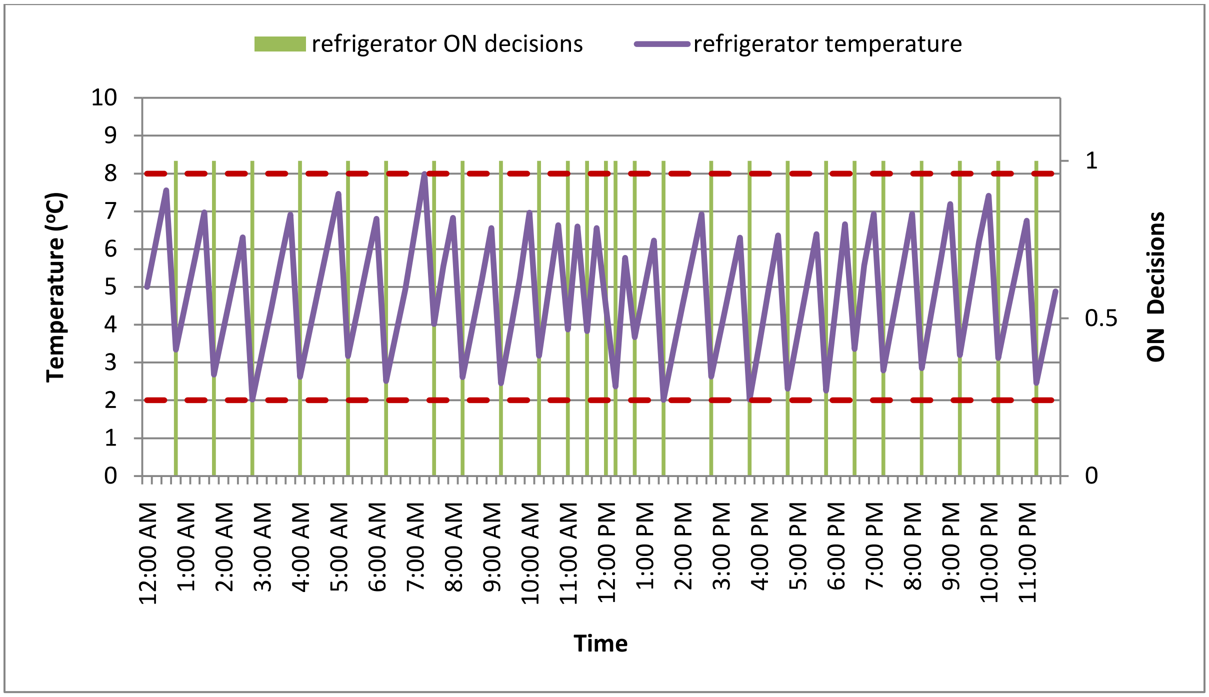


Figure 3.12. Refrigerator temperature and ON decisions.

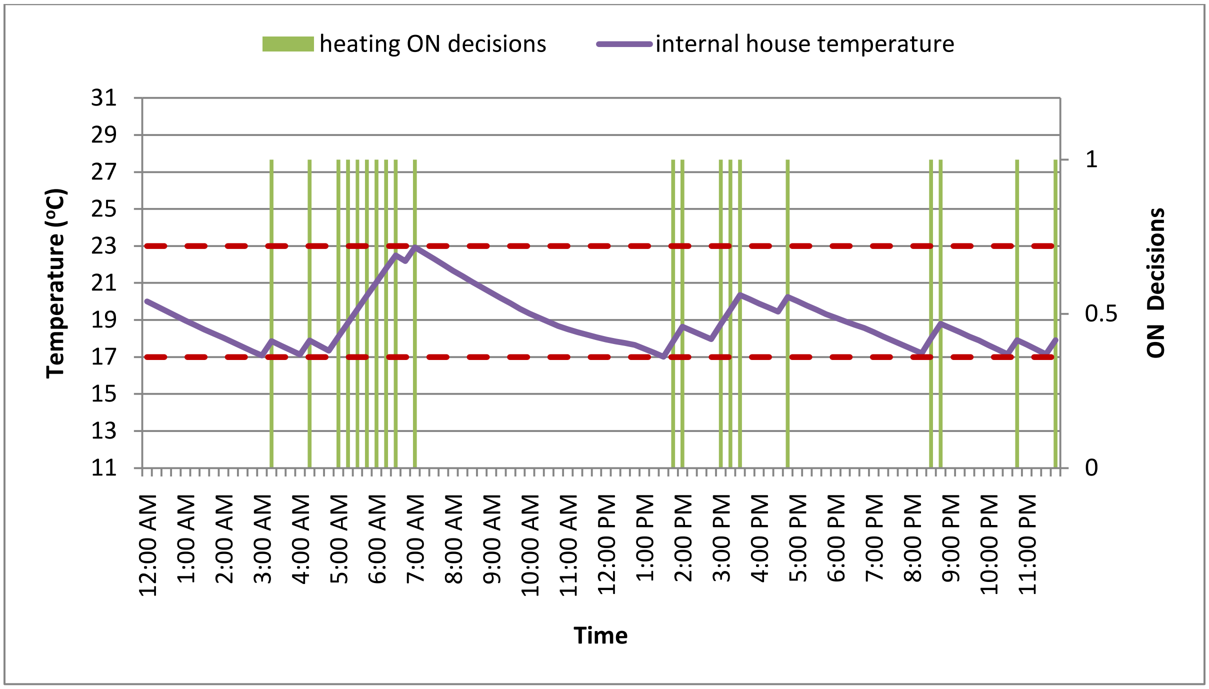


Figure 3.13. Heating system ON decisions and corresponding house temperature.

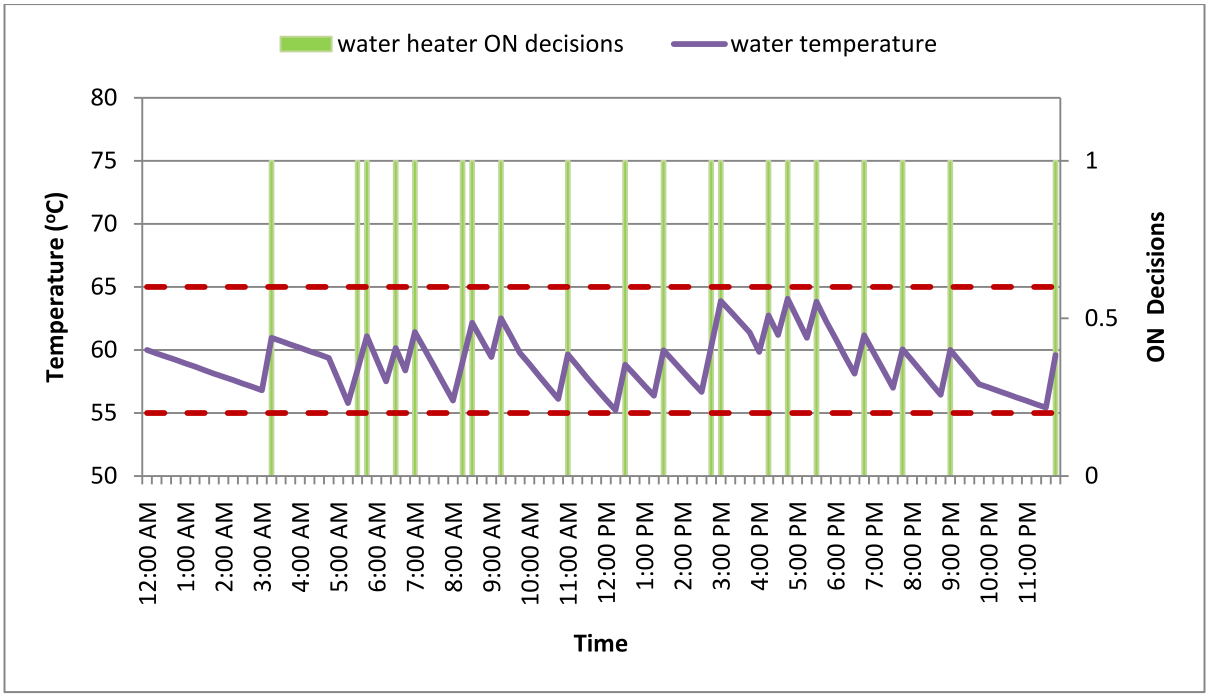


Figure 3.14. Water heater temperature and ON decisions.

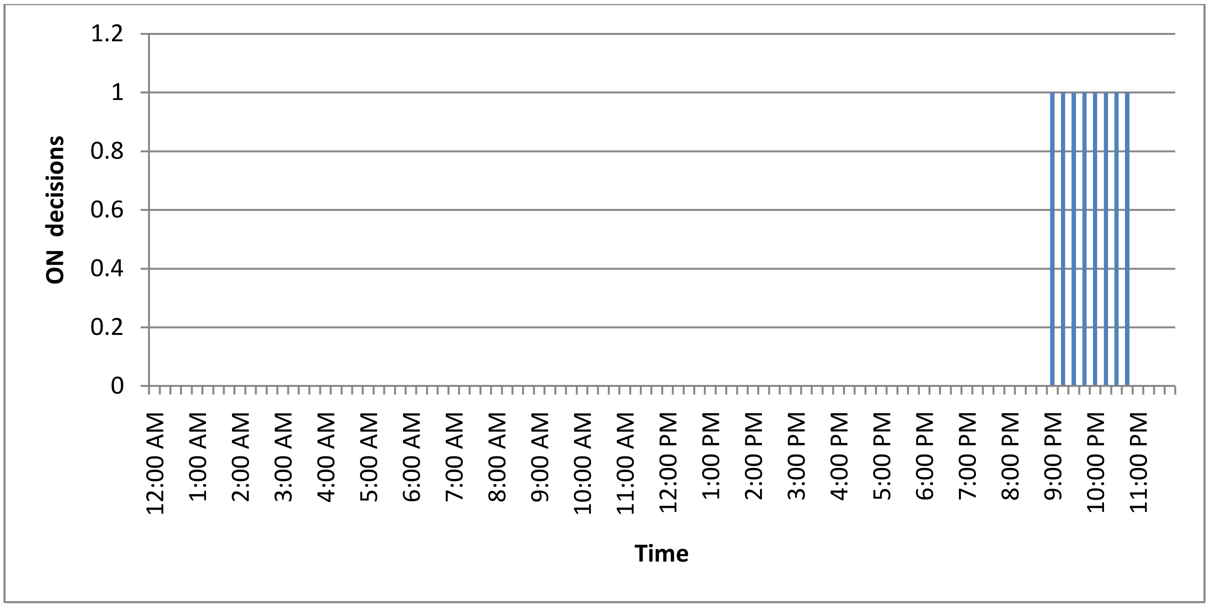


Figure 3.15. Dishwasher ON decisions.

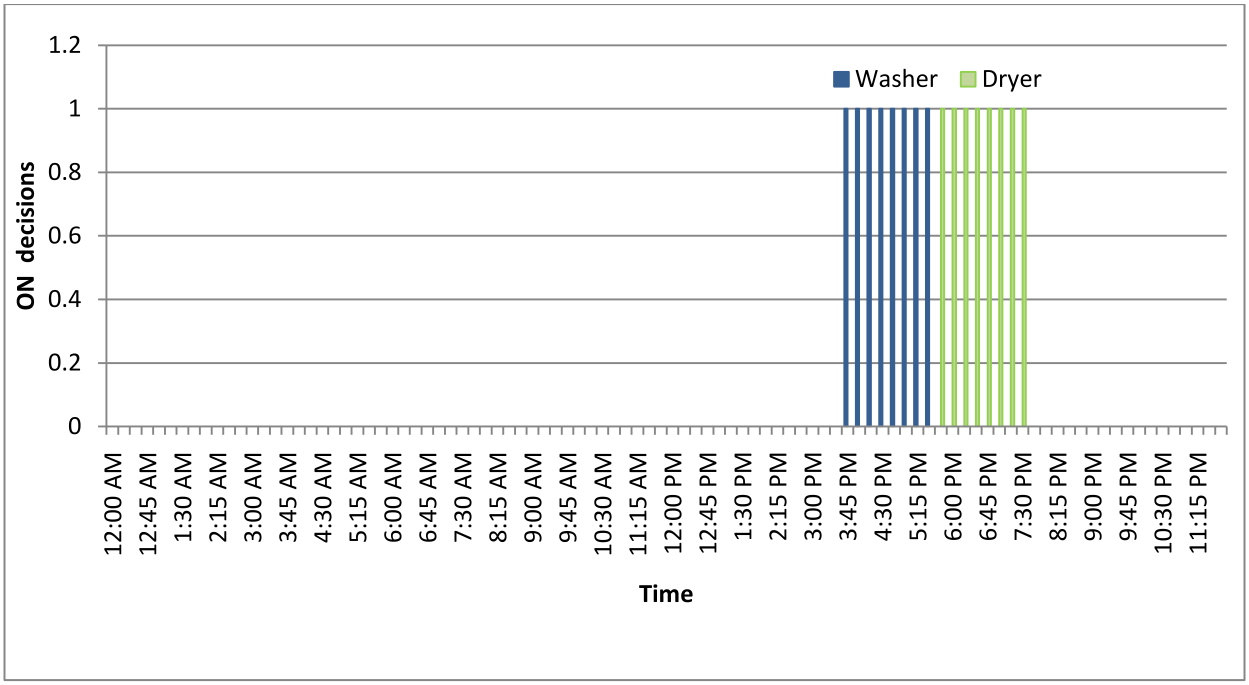


Figure 3.16. Cloth washer and dryer ON decisions.

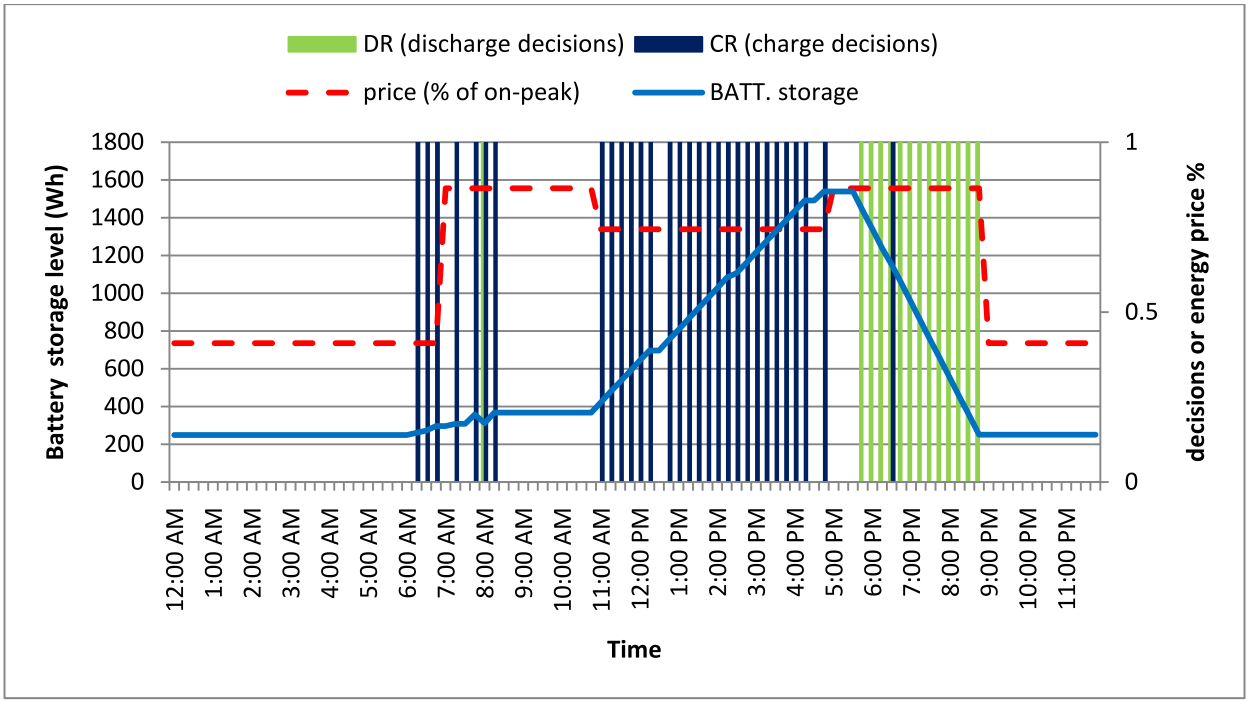


Figure 3.17. PV system battery charging/discharging decisions and battery storage level.

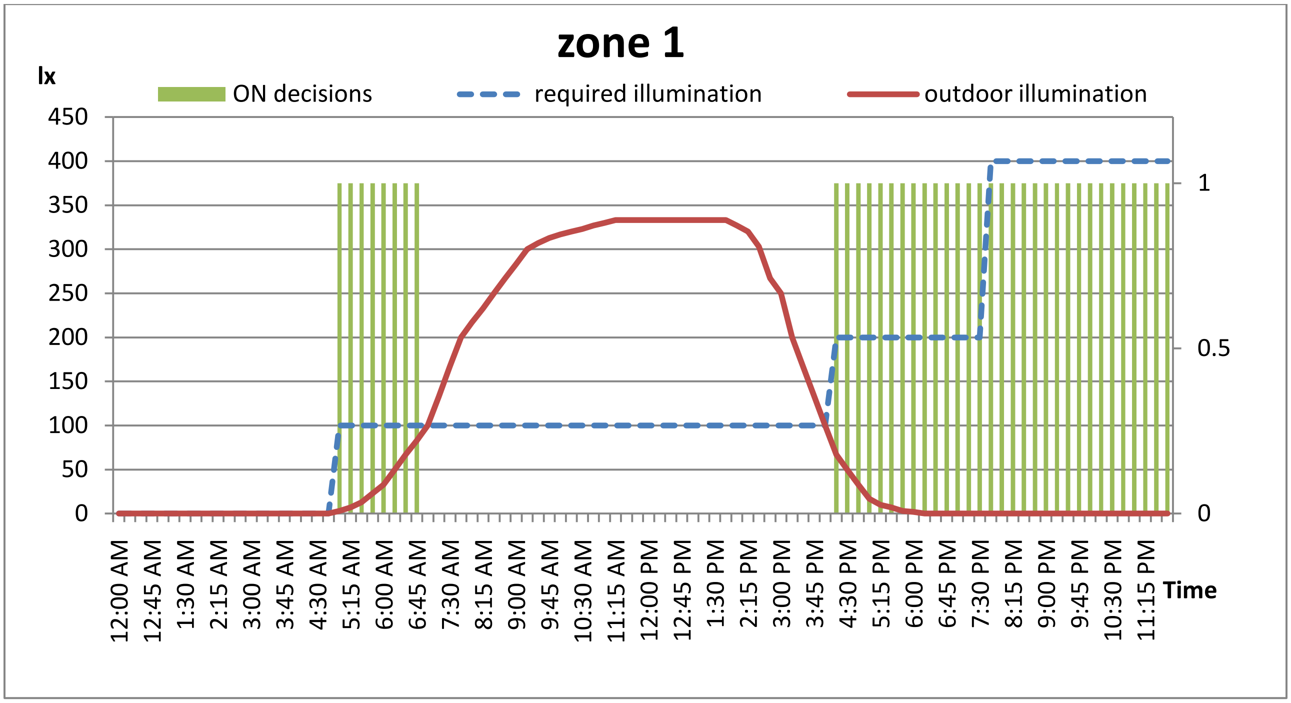


Figure 3.18. Illumination level and lighting ON decisions of zone 1.

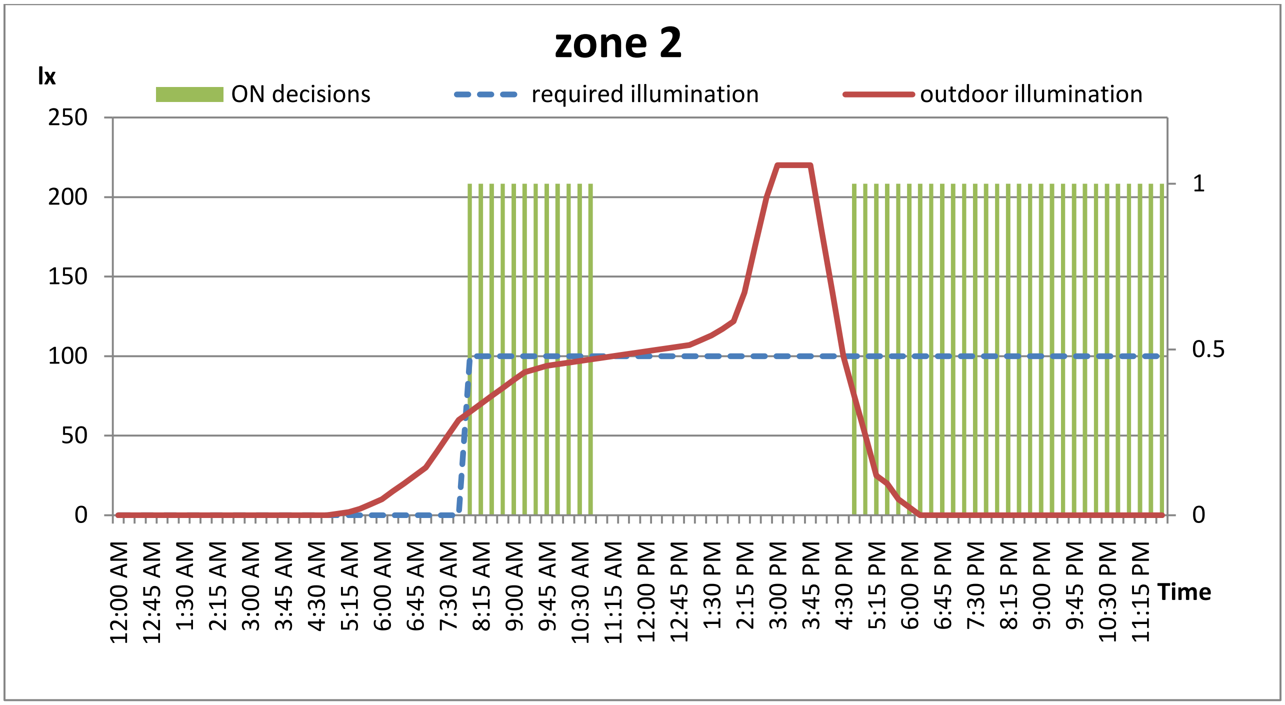


Figure 3.19. Illumination level and lighting ON decisions of zone 2.

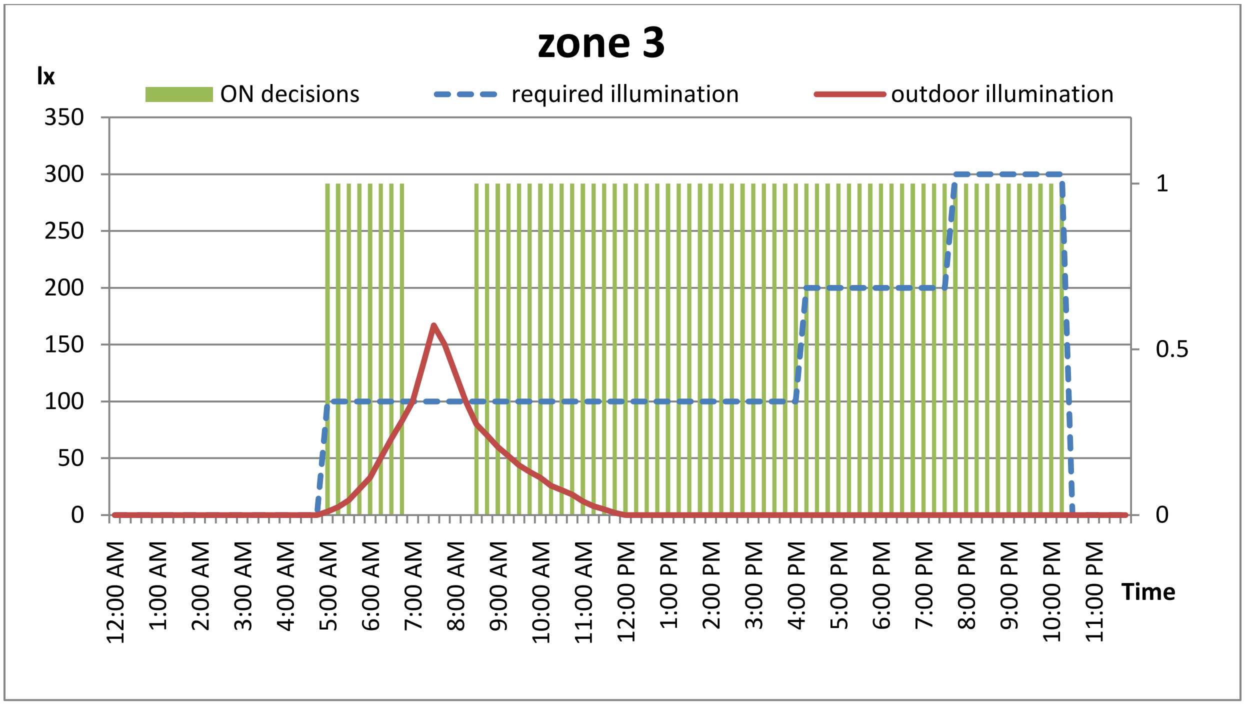


Figure 3.20. Illumination level and lighting ON decisions of zone 3.

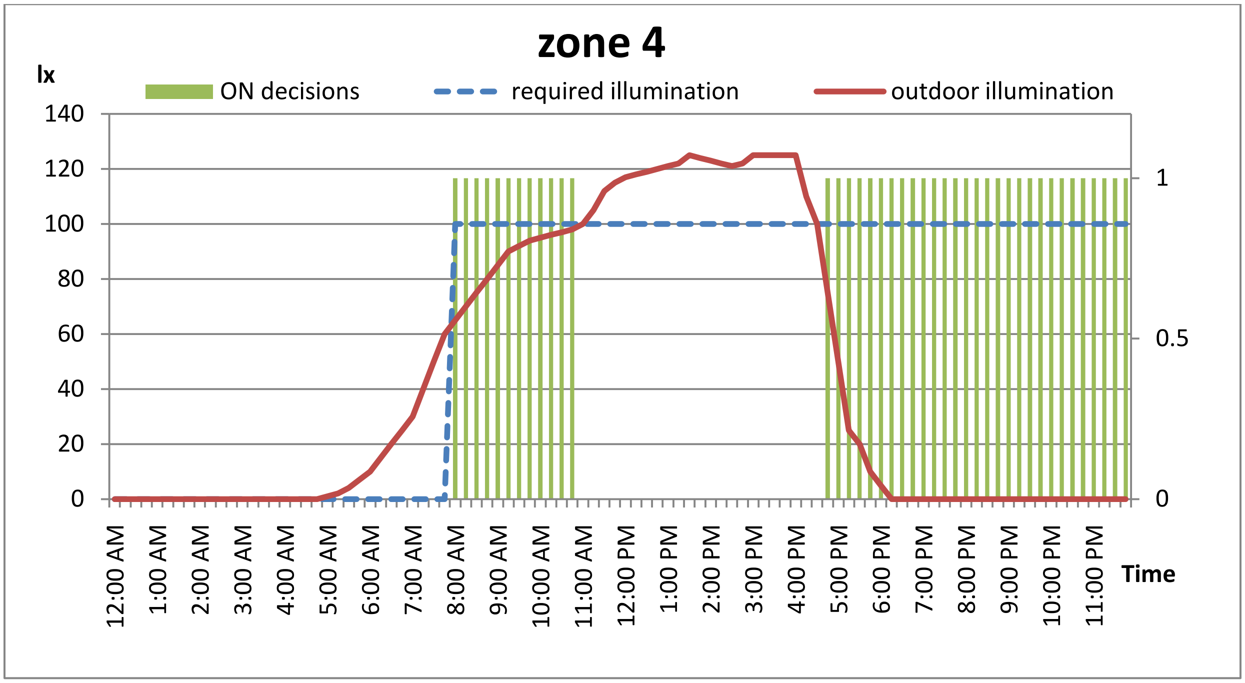


Figure 3.21. Illumination level and lighting ON decisions of zone 4.

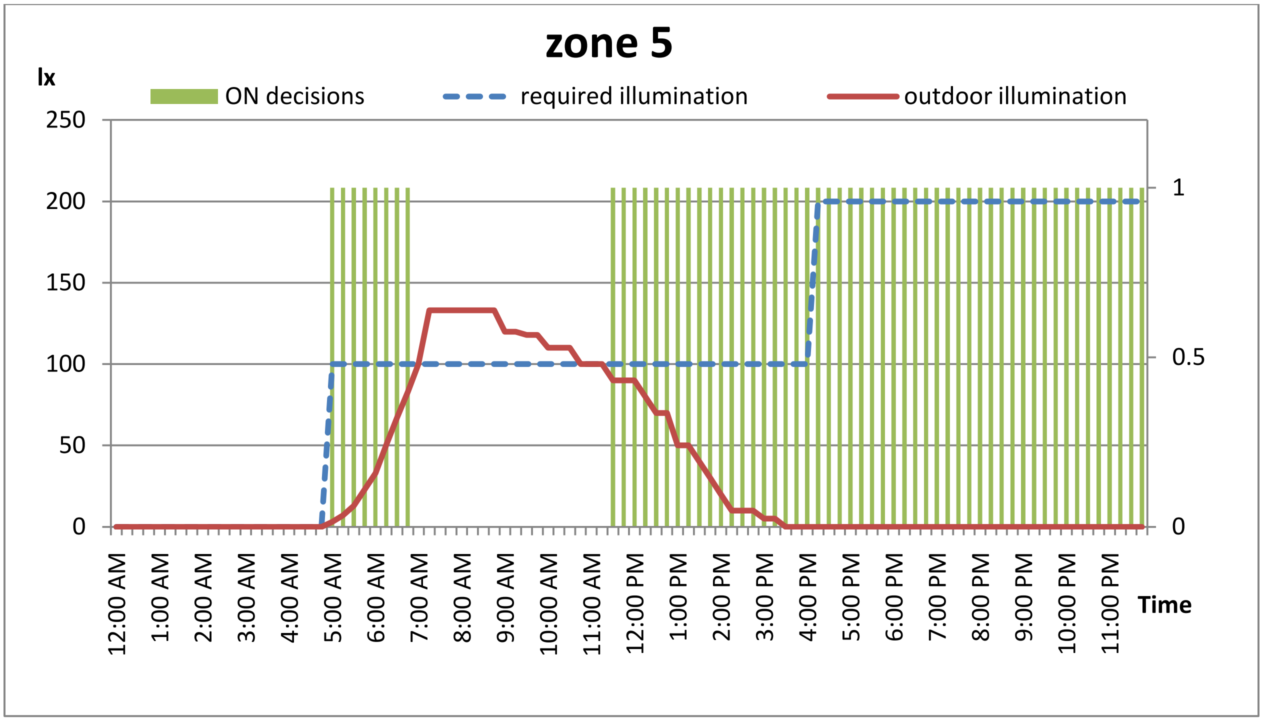


Figure 3.22. Illumination level and lighting ON decisions of zone 5.

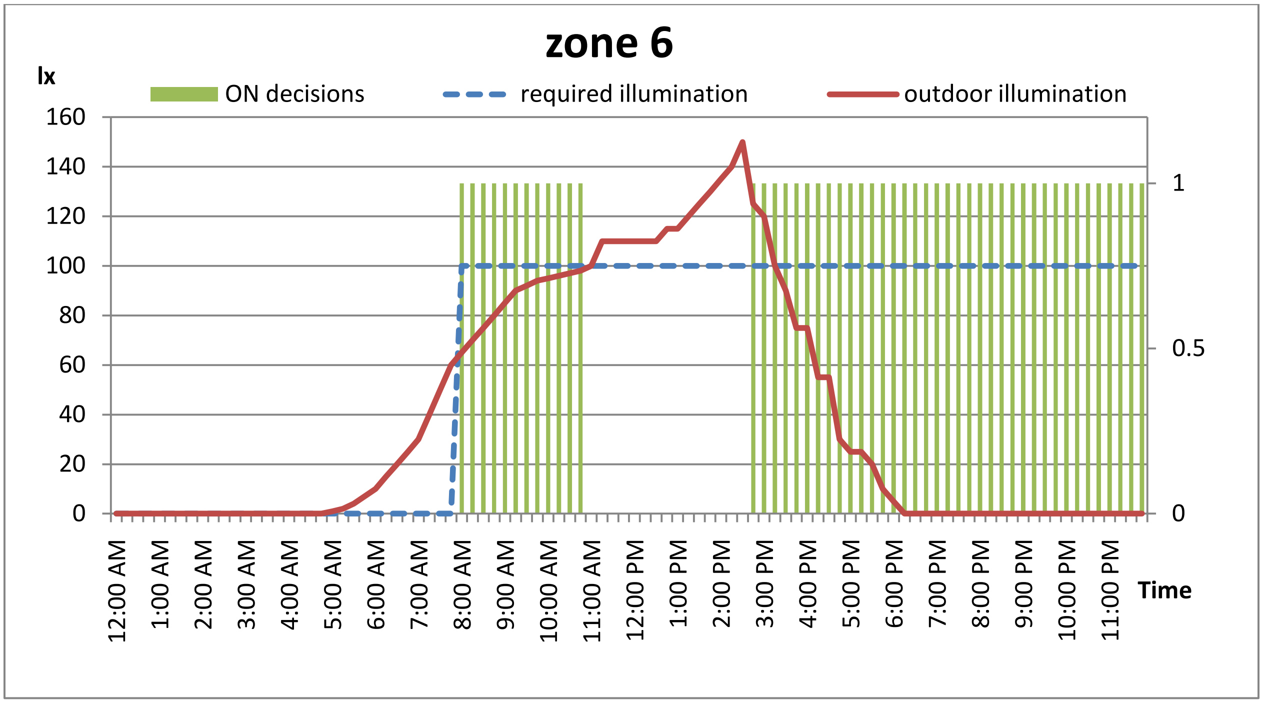


Figure 3.23. Illumination level and lighting ON decisions of zone 6.

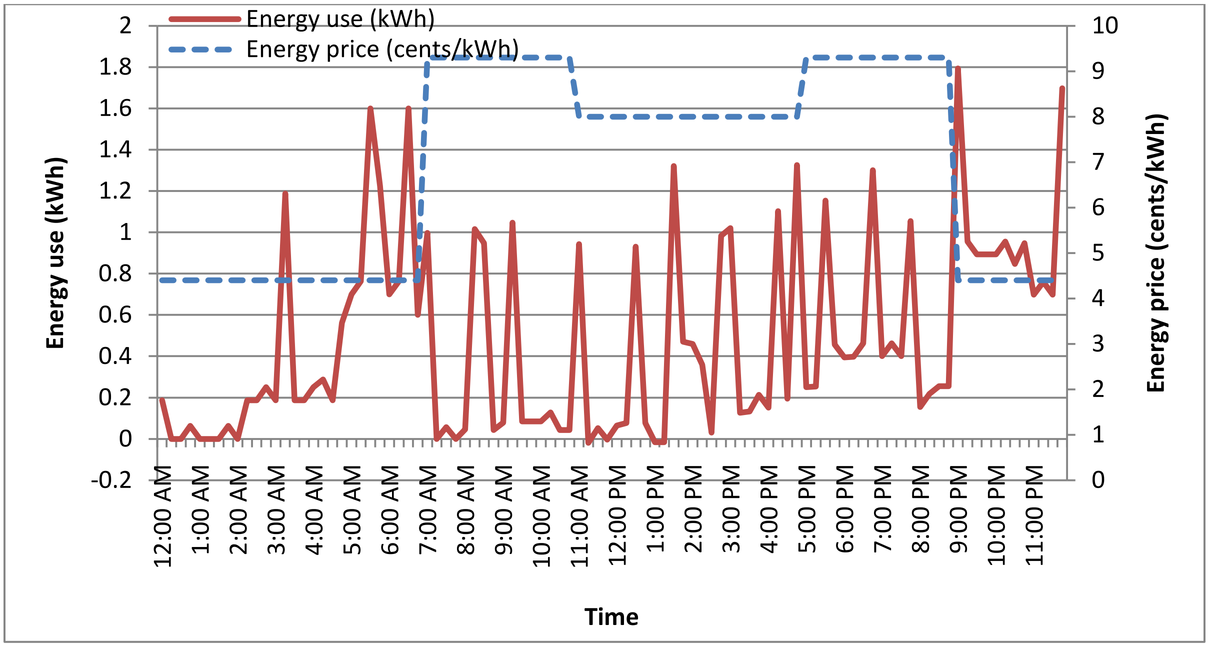


Figure 3.24. Household electrical energy consumption.

3.2 Experimental Validation- Refrigerator Case Study

3.2.1 No Door Activity

An experiment was set up to show the practical determination of the $\alpha_{FR}, \beta_{FR}\ and\ \gamma_{FR}$ parameters and to validate the mathematical model of the refrigerator. A small 4.3 cb.ft refrigerator was used for these purposes, and it was loaded with twenty 0.5 liter water bottles to simulate the presence of food inside the refrigerator.

To measure α_{FR} and γ_{FR} the temperature inside the refrigerator was measured using a temperature data logger type Omega OM-62. The logger was kept in the center of the refrigerator cabinet and it was set to log temperature measurements every minute. The electric power consumption of the refrigerator was also logged at the same time using an EXTECH Energy Logger model EM100, which was set to log compressor input power consumption every minute as well. Measuring devices are shown in Figure 3.25.

Energy Data Logger

Digital Programmable Timer to simulate the EHMS ON/OFF decisions

Temperature Data Logger

Figure 3.25. Measuring and control devices used in the experiment.

The logged events are shown in Figure 3.26, from which it can be seen that the refrigerator temperature rises by 0.081^{o}C/min when the compressor is OFF. From this one can calculate $\gamma_{FR} = 0.081\frac{℃}{\min} \times 15\ \min = 1.21$^{o}C per 15-minute time interval. When the compressor is ON, the temperature drops by 0.285^{o}C/min; this drop is a result of temperature drop due to compressor operation minus the temperature rise due to ambient heat entering the refrigerator through shield and gasket. Hence the temperature drop due to compressor operation only is equal to: 0.285^{o}C/min + 0.081^{o}C/min = 0.366^{o}C/min. Accordingly, $\alpha_{FR} = 0.366\frac{℃}{\min} \times 15\ \min = 5.5$^{o}C for the case of 15-minute time intervals. The empirical values of α_{FR} and γ_{FR} were implemented in calculating the optimal refrigerator decisions using the GLPK solver, which produced the results shown in Table 3.6.

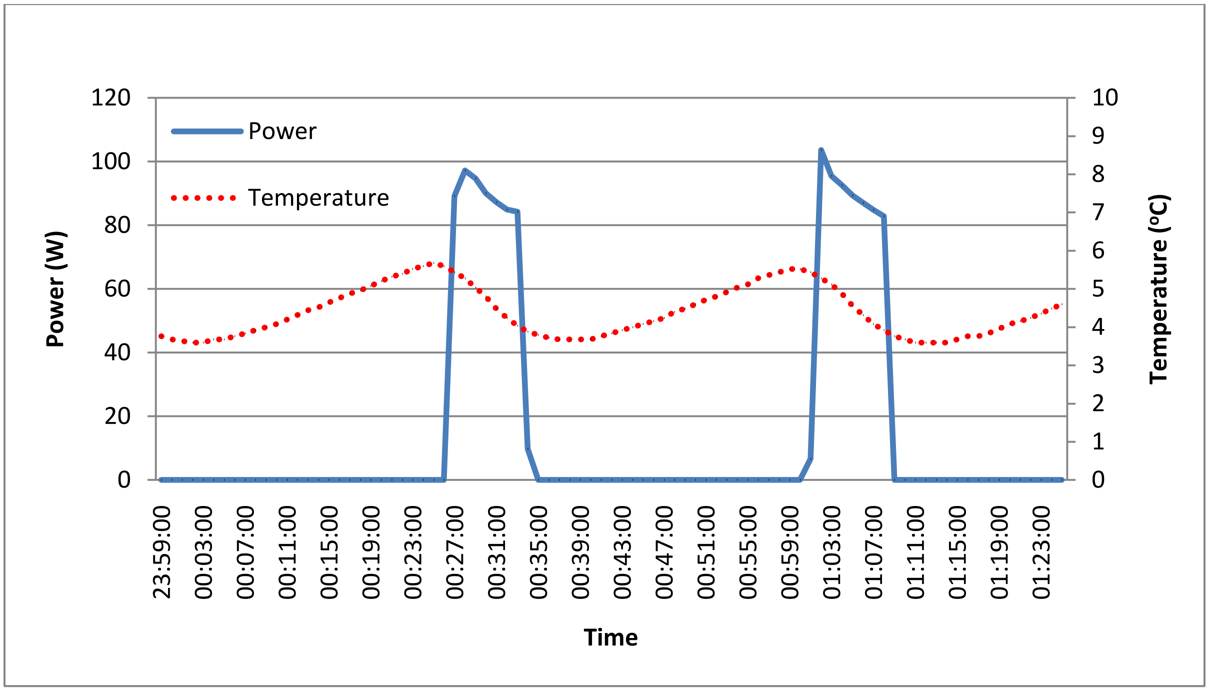


Figure 3.26. Refrigerator power consumption and temperature.

Table 3.6. GLPK solver optimum decisions.

Schedule	Time	Decision	Temperature (°C)	Schedule	Time	Decision	Temperature (°C)	Schedule	Time	Decision	Temperature (°C)
1	12:00:00 AM	0	4	33	8:00:00 AM	0	5.34	65	4:00:00 PM	1	1.38
2	12:15:00 AM	0	5.2	34	8:15:00 AM	1	1.24	66	4:15:00 PM	0	2.58
3	12:30:00 AM	1	1.11	35	8:30:00 AM	0	2.44	67	4:30:00 PM	0	3.78
4	12:45:00 AM	0	2.31	36	8:45:00 AM	0	3.64	68	4:45:00 PM	0	4.98
5	1:00:00 AM	0	3.51	37	9:00:00 AM	0	4.84	69	5:00:00 PM	0	6.18
6	1:15:00 AM	0	4.71	38	9:15:00 AM	0	6.04	70	5:15:00 PM	1	2.08
7	1:30:00 AM	0	5.91	39	9:30:00 AM	1	1.95	71	5:30:00 PM	0	3.28
8	1:45:00 AM	1	1.81	40	9:45:00 AM	0	3.15	72	5:45:00 PM	0	4.48
9	2:00:00 AM	0	3.01	41	10:00:00 AM	0	4.35	73	6:00:00 PM	0	5.68
10	2:15:00 AM	0	4.21	42	10:15:00 AM	0	5.55	74	6:15:00 PM	1	1.59
11	2:30:00 AM	0	5.41	43	10:30:00 AM	0	6.75	75	6:30:00 PM	0	2.79
12	2:45:00 AM	1	1.32	44	10:45:00 AM	1	2.65	76	6:45:00 PM	0	3.99
13	3:00:00 AM	0	2.52	45	11:00:00 AM	0	3.85	77	7:00:00 PM	0	5.19
14	3:15:00 AM	0	3.72	46	11:15:00 AM	0	5.05	78	7:15:00 PM	1	1.09
15	3:30:00 AM	0	4.92	47	11:30:00 AM	0	6.25	79	7:30:00 PM	0	2.29
16	3:45:00 AM	0	6.12	48	11:45:00 AM	1	2.16	80	7:45:00 PM	0	3.49
17	4:00:00 AM	1	2.02	49	12:00:00 PM	0	3.36	81	8:00:00 PM	0	4.69
18	4:15:00 AM	0	3.22	50	12:15:00 PM	0	4.56	82	8:15:00 PM	0	5.89
19	4:30:00 AM	0	4.42	51	12:30:00 PM	0	5.76	83	8:30:00 PM	1	1.8
20	4:45:00 AM	0	5.62	52	12:45:00 PM	1	1.66	84	8:45:00 PM	0	3
21	5:00:00 AM	0	6.82	53	1:00:00 PM	0	2.86	85	9:00:00 PM	0	4.2
22	5:15:00 AM	1	2.73	54	1:15:00 PM	0	4.06	86	9:15:00 PM	0	5.4
23	5:30:00 AM	0	3.93	55	1:30:00 PM	0	5.26	87	9:30:00 PM	1	1.3
24	5:45:00 AM	0	5.13	56	1:45:00 PM	1	1.17	88	9:45:00 PM	0	2.5
25	6:00:00 AM	1	1.03	57	2:00:00 PM	0	2.37	89	10:00:00 PM	0	3.7
26	6:15:00 AM	0	2.23	58	2:15:00 PM	0	3.57	90	10:15:00 PM	0	4.9
27	6:30:00 AM	0	3.43	59	2:30:00 PM	0	4.77	91	10:30:00 PM	0	6.1
28	6:45:00 AM	0	4.63	60	2:45:00 PM	0	5.97	92	10:45:00 PM	1	2.01
29	7:00:00 AM	0	5.83	61	3:00:00 PM	1	1.87	93	11:00:00 PM	0	3.21
30	7:15:00 AM	1	1.74	62	3:15:00 PM	0	3.07	94	11:15:00 PM	0	4.41
31	7:30:00 AM	0	2.94	63	3:30:00 PM	0	4.27	95	11:30:00 PM	0	5.61
32	7:45:00 AM	0	4.14	64	3:45:00 PM	0	5.47	96	11:45:00 PM	0	6.81

In order to implement the optimum decisions produced by GLPK, a digital programmable timer type “KILL AWATT”, was used. This timer has 15-minute time resolution and 96 different program settings per day. The ON/OFF schedules of the digital timer were set according to the optimum decisions produced by GLPK MILP solver to simulate the actual case of the EHMS decisions. The refrigerator thermostat setting was set to MINIMUM (clockwise direction) in order to keep the thermostat always ON, so that it will not interrupt the optimal ON/OFF decisions programmed by the digital programmable timer.

The results of the measured experimental data, which are shown in Figure 3.27, show that the measured temperature is shifted up over the GLPK calculated data by about 1.3°C. The reason for this is the error in estimating the initial temperature data used in the GLPK optimization code. In practice, this error will not happen because the solver will receive the data from the temperature sensor inside the refrigerator every 15 minutes, and solve the optimization code in about 60 s, yielding the optimum ON/OFF decisions.

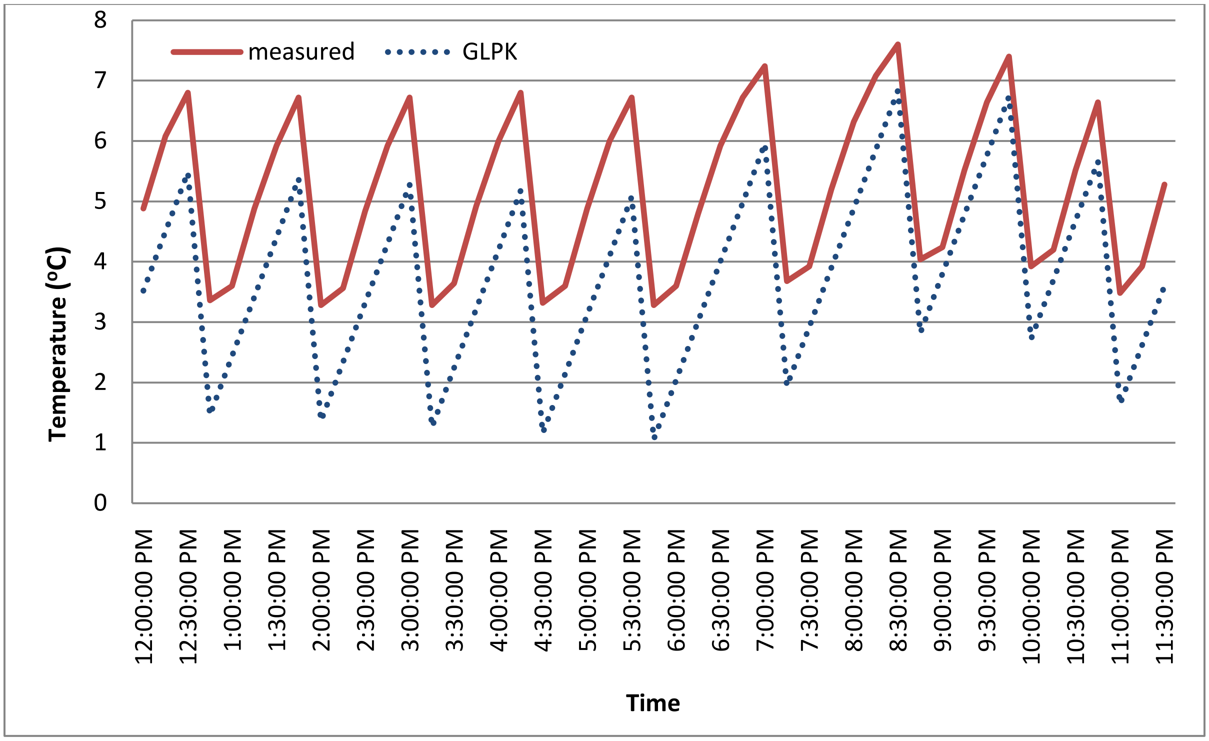


Figure 3.27. Measured and GLPK calculated refrigerator temperatures.

3.2.2 With Door Activity

In this experiment the refrigerator door was opened 30 times for approximately 10 s each time, according to the schedule shown in Table 3.7, which results in the activity level shown in Figure 3.28. GLPK was used to calculate the optimum compressor ON/OFF decisions and the digital programmable timer was used again to implement these decisions.

Table 3.7. Refrigerator door activity schedule.

morning activity	number of door opening	Activity Level (p.u.)
09:30 AM - 09:45 AM	2	0.07
09:45 AM - 10:00 AM	3	0.1
10:00 AM - 10:15 AM	4	0.13
10:15 AM - 10:30 AM	1	0.03
afternoon activity		
01:30 PM - 01:45 PM	2	0.07
01:45 PM - 02:00 PM	5	0.17
02:00 PM - 02:15 PM	7	0.23
02:15 PM - 02:30 PM	4	0.13
02:30 PM - 02:45 PM	2	0.07

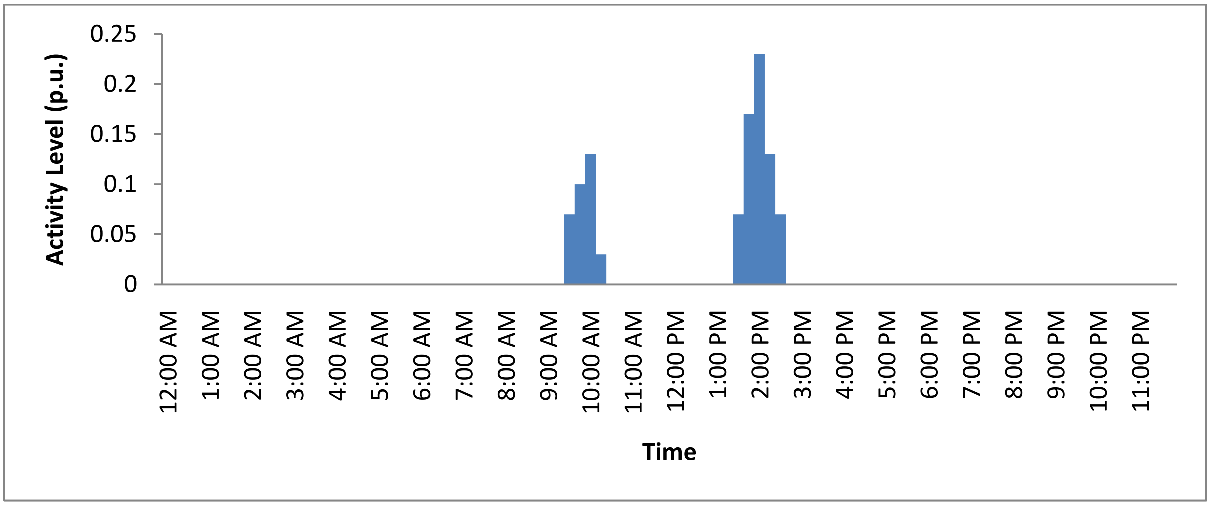


Figure 3.28. Door opening activity level.

The temperature was measured and compared to the calculated temperature by GLPK, and the results are shown in Figure 3.29. Observe that the measured temperatures are shifted above the GLPK calculated temperatures due to the misestimation of the initial temperatures, which is not expected to take place in the practical implementation of the EHMS. For the purpose of comparison, Figure 3.30 shows the refrigerator temperature with the same door opening activity under thermostat control; observe how the temperature is raised up to about 3^{o}C above the thermostat set point of 5 ^{o}C during the door open activity, whereas the proposed model keeps the temperatures within the defined range.

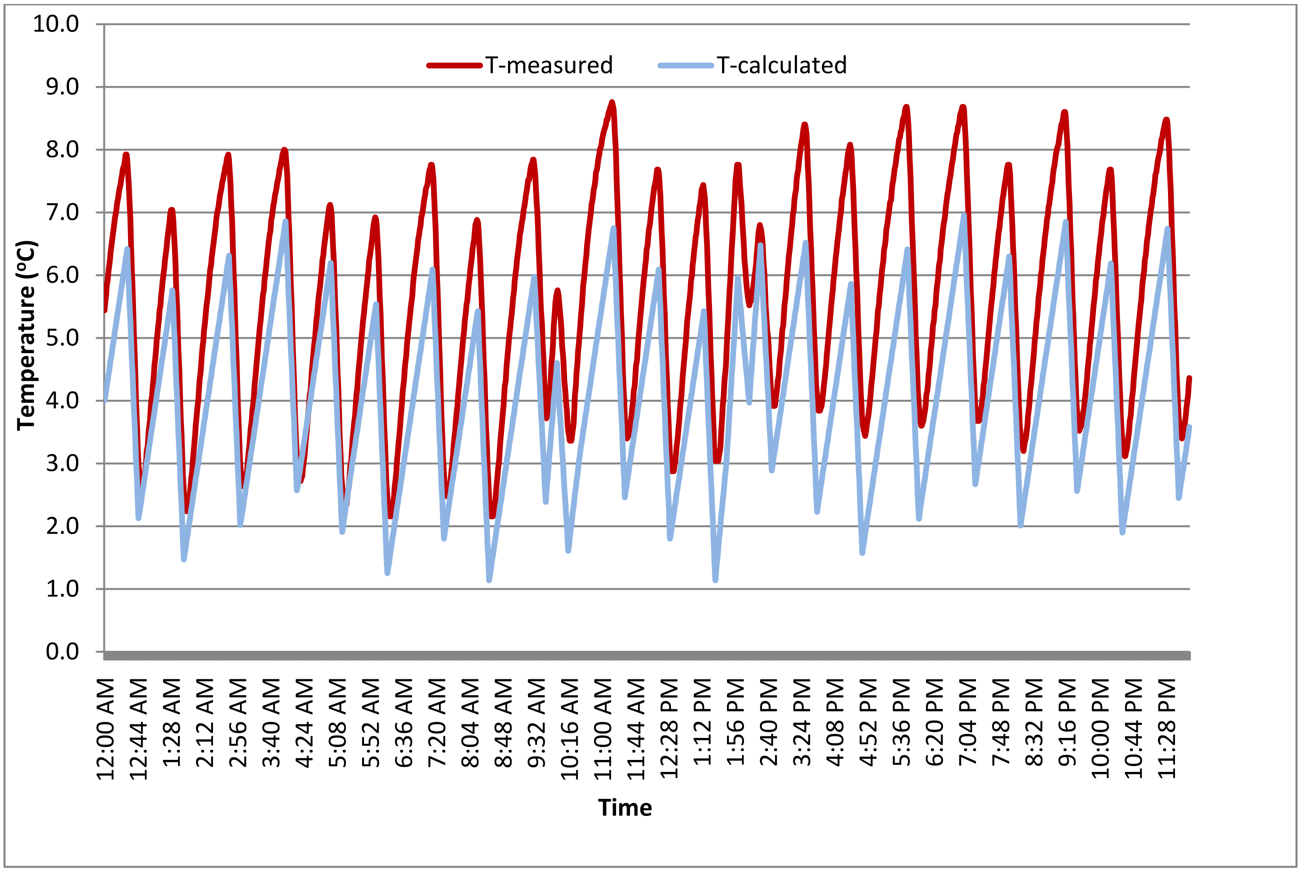


Figure 3.29. Measured and GLPK calculated temperatures.

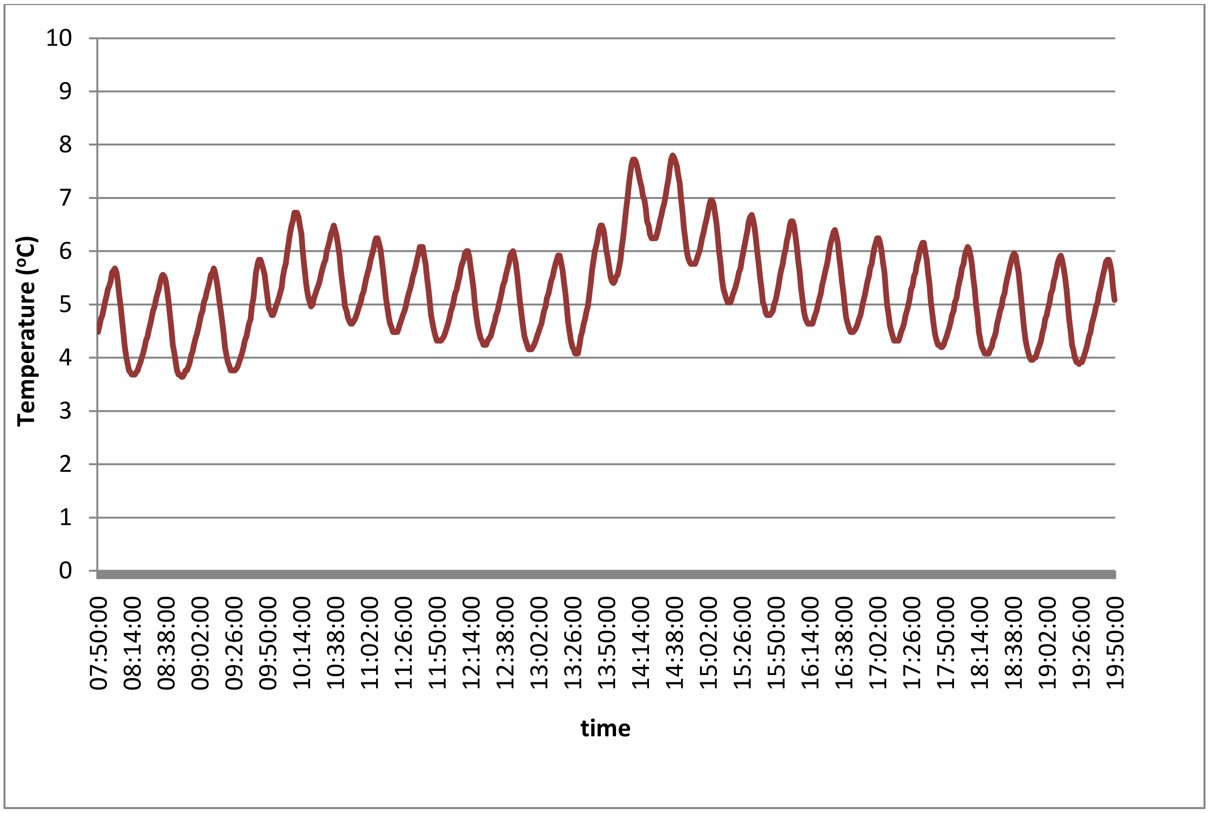


Figure 3.30. Refrigerator under thermostat control with door opening activity.

3.3 Summary

This chapter started with a brief introduction of the GLPK solver, which was then used to solve the mathematical model of a sample house. The code, originally written in AMPL and solved using CPLEX in [9], was modified in this work into two versions: CDMRes4A (see Appendix A) and CDMR4B (see Appendix B), with minor and major modifications respectively, based on the discussions presented in Chapter 2. The parameters used for each model were presented and the model optimization results data were obtained and discussed. Finally, the results of a real time experiment on the refrigerator were discussed. The GLPK solver proved to be reliable for use in the residential EHMS, and the experiment results show that the mathematical models and parameters measurements are valid.

Chapter 4

Conclusions and Future Works

4.1 Summary and Conclusions

This thesis examines the potential implementation aspects of the mathematical models of the residential appliances for the EHMS. The model of each appliance has been intensely investigated in detail and the parameters of each model have been determined either through measurements or using thermodynamic equations. Improvements or simplifications have been proposed for some models, and a mathematical model of a PV solar system has been proposed and demonstrated. The best options for the chosen GLPK MILP solver have been also identified, and finally, the mathematical model of the refrigerator has been validated through a practical experiment.

The generic residential DSM mathematical model proposed in [9] presents a new and a promising method for load control, because it integrates all different appliances of different behaviors into a unified optimization model. However, its implementation faces many obstacles such as the need for a high speed MILP solver and the availability of two-way ON/OFF actuators. Although the GLPK freeware solver proved to provide enough computational speed so far, this aspect should be further investigated for other applications such as commercial and industrial sectors. In order to solve the problem in a real time, simplifications for the mathematical model for each appliance are necessary, which will affect the accuracy of the results. From the work presented here, implementation of the EHMS mathematical models for the residential appliances seems feasible and useful, but these mathematical models should be further developed and improved as more data on appliances' operation and customer behavior and preferences are collected during the implementation process.

4.2 Contributions

There are several contributions in this thesis which are summarized as follows:

1- Developing procedures to calculate or estimate the parameters of the residential appliance mathematical models proposed in [9] based on either measurements using data loggers or by calculation using thermodynamic equations.

2- Proposing a mathematical model for the domestic PV system to be used within the EHMS.

3- Proposing simplifications for the models of dishwashers, washers and dryers in order to enable the use of 96 15-minute time intervals for all of domestic appliances to improve the computational efficiency.

4- Investigating the use of the GLPK solver to determine its performance and the best options which can lead to shorter problem solving time.

5- Testing and validation of the mathematical model of the refrigerator through a real time experiment and measuring the variables and parameters using data loggers.

4.3 Future Work

Although some improvement has been proposed for some appliances, the cooling/heating model should be improved further by including the effect of solar gain and infiltrations of the house temperature and energy consumption. Also, a detailed cost analysis is required to estimate the yearly cost savings for the appliances to be implemented in the EHMS. At the same time, a detailed analysis is required to estimate the aggregated percentage of load shifting generated from controlling each appliance.

More research is required to improve the estimation of activity level of each appliance and the forecasting of outdoor temperature and daylight illumination. Collection of more data via additional experiments or several household monitoring will be helpful in this regard. Finally, the estimation of hot water use being the extended EPRI model mentioned in this thesis should be investigated for its suitability in Canadian households.

Appendix A

Code CDMRes4A.mod

```
############################################################################
# By Mohammad Chehreghani Bozchalui @ Nov    2009                          #
# modefied by Hussin Hassen on 19 Feb. 2010                                #
# @ University of Waterloo                                                 #
#                                                                          #
############################################################################
#reset;
set ADEVICES;                               # set of devices work on 96 time schedules
set SCHEDULES;                               # set of 96 time schedules
set BSCH;                                    # set of devices work on 24 time schedules
set BDEVICES;                                # set of devices work on 24 time schedules
################
# A-Parameters #
################

param PRICE {SCHEDULES};                      # Price per schedule in "cent/kwh"
param POWER {ADEVICES union BDEVICES};        # Power cunsumption of each Device in "kw"
param POWERCONS {SCHEDULES};                  # Total power consumption at each time schedule

param EEC {j in SCHEDULES};     # ELECTRICAL ENERGY CONSUMPTION in KWhr(of the previous week)
param EEC1{j in SCHEDULES}:= EEC[j] less (0.5 * (sum {k in 1..96} EEC[k])/96);
                                              #Dummy parameter for calculating ActLev
param ActLevH {j in SCHEDULES}:= (100*EEC [j]) / sum {k in 1..96} EEC[k] ;
                                         # HEATING Activity Level at each hr of time schedule
param ActLevF {j in SCHEDULES}:= (100*EEC1[j]) / sum {k in 1..96} EEC1[k];
                                       # FRG usage Activity Level at each hr of time schedule

param OutIllum {SCHEDULES};           # daylight Illumination, which entered from outside
param ReqIllum {SCHEDULES};                   # minimum required Illumination

param TempFrigUP;                             # Fridge Temperature Up limit
param TempFrigDN;                             # Fridge Temperature Down limit
param TempFrigIC;                             # Fridge Temperature Initial Temp
param BetaFrig; # ActLev Coefficient in the TempFrig formula.
param AlfaFrig;# Cooling effect coeff. of previous ON time interval in the TempFrig formula
param GamaFrig;# Warming effect coeff. of previous OFF time interval in the TempFrig formula

param TempWaterheaterUP;                      # Waterheater Temperature Up limit
param TempWaterheaterDN;
param TempWaterheaterIC;
param BetaWaterheater ;
param AlfaWaterheater;
param GamaWaterheater;

param AlfaHeating;
param TempINUP;                               # Indoor Temperature Up limit
param TempINDN;                               # Indoor Temperature Down limit
param TempIC;                                 # indoor starting initial temparature
param OutTemp {SCHEDULES};                    # Outdoor Temperature at each hr of time
schedule (measured)
param BetaAC;                                 # ActLev Coefficient in the TempIN formula.
param AlfaAC;  # Cooling effect coeff. of previous ON time interval in the TempIN formula
param LambAC;  # the effect of outdoor temp. of previous ime interval in the TempIN formula

param PoolpumpROT;                            # Poolpump Req. Operation Time
param PoolpumpEarlTime;
param PoolpumpLateTime;

param ESDEarlTime;                             # Energy Storage Device
param ESDLateTime;
```

```
param ESDChargeRate{SCHEDULES};# ESD generation rate at time interval t, in kw/15min.
param ESDMinStorage;
param ESDMaxStorage;
param StorESD0;


param StoveMinUP;
param StovMSOT;
param StovROT;                                # Stov Req. Operation Time

param  LightEarlTime;#Earliest Possible Scheduling time for Lighting (as per user preference)
param  LightLateTime;#Latest Possible Scheduling time for Lighting (as per  user preference)
param  ACEarlTime;
param  ACLateTime;
param  WaterheaterEarlTime;
param  WaterheaterLateTime;
param  FridgeEarlTime;
param  FridgeLateTime;
param  StoveEarlTime;
param  StoveLateTime;
param  TWHEarlTime;
param  TWHLateTime;

################
# B-Parameters #
################

param   BPRICE {BSCH};                      # BPRICE per schedule in "cent/kwh"

param BEEC {j in BSCH};# ELECTRICAL ENERGY CONSUMPTION in KWhr(of the previous week)
param BEEC1{j in BSCH}:= BEEC[j] less (0.5 * (sum {k in 1..24} BEEC[k])/24);
                                                   #Dummy parameter for calculating ActLev
param BActLevD {j in BSCH}:= (100*BEEC1[j]) / sum {k in 1..24} BEEC1[k];
                                     # Dishwasher Activity Level at each hr of time schedule

param  DishwasherEarlTime;
param  DishwasherLateTime;
param  DishwasherMinUP;
param  DishwasherMinDN;

param  WasherEarlTime;
param  WasherLateTime;
param  WasherMinUP;
param  WasherMinDN;
param   person;                                # number of persons living in a  a house

param  DryerEarlTime;
param  DryerLateTime;

param   StorLevDW0;                          # Initial Storage level of Dishwasher in "kwh"
param   BetaDishwasher;       # Effect of BActLev on Dishwasher Storage at each time interval
param   CapDishwasher;                       # Storage Capacity of Dishwasher in "kwh"

param   StorLevWasher0;                # Initial Storage level of Washer in "kwh"
param   CapWasher;                           # Storage Capacity of Wsher in "kwh"

param   StorLevDryer0;                       # Initial Storage level of Dryer in "kwh"
param   CapDryer;                            # Storage Capacity of Dryer in "kwh"

param   MTGapWashDryer;# Maximum Allowable Time Gap between operations of Washer and Dryer

#################################################################
# additional parameters for calculation of hourly hot water use #
#################################################################

param  per ;    # number of persons in a house,
param  age1;    # number of preschool children (0-5 yrs)
```

```
param  age2;    # number of school age children (6-13 yrs)
param  age3;    # number of adults (14 yrs and over)
param  therm;   # water heater thermostat setting, (°C)
param  tankz;   # water heater nominal tank size, (Liters)
param  wtmp;    # water heater inlet water temperature, (°C)
param  atmp;    # outdoor air ambient temperature, (°C)
param  athome;  # a dummy param for the presence of unemployed person at home during day (
param  spring;  # a dummy parameter for Spring (1 if “spring”, zero otherwise)
param  summer;  # a dummy parameter for Summer (1 if “spring”, zero otherwise)
param  fall;    # a dummy parameter for fall (1 if “spring”, zero otherwise)
param  winter;  # a dummy parameter for winter (1 if “spring”, zero otherwise)
param  noDW;    # a dummy parameter (1 if dishwasher is not available, zero otherwise)
param  noCW;    # a dummy parameter (1 if washer is not available, zero otherwise)
param  noDWeq{j in SCHEDULES}:= if (65 <=j and j<= 80) then (0.5-atmp/80)*(2.62*per+5.054*
sqrt(per))/4 else 0; #a dummy function indicating impact of not owning a dish washer
param  noCWeq{j in SCHEDULES}:= if (65 <=j and j<= 80) then (0.5-atmp/80)*(4.4242*per+18.070*
sqrt(per))/4 else 0; #a dummy function indicating impact of not owning a clothes washer
param  senior;  # a dummy parameter for senior = 1, non-senior = 0 otherwise
param  nopay;   # a dummy parameter = 1 if household does not pay for hot water,otherwise 0
param  a0{SCHEDULES};
param  a1{SCHEDULES};
param  a2{SCHEDULES};
param  a3{SCHEDULES};
param  a4{SCHEDULES};
param  a5{SCHEDULES};
param  a6{SCHEDULES};
param  a7{SCHEDULES};
param  a8{SCHEDULES};
param  a9{SCHEDULES};
param  a10{SCHEDULES};
param  a11{SCHEDULES};
param  a12{SCHEDULES};
param  a13{SCHEDULES};
param HWL{j in SCHEDULES}:=
(a0[j]+a1[j]*per+a2[j]*age1+a3[j]*age2+a4[j]*age3+a5[j]*therm+a6[j]*tankz+a7[j]*wtmp+a8[j]*at
mp+a9[j]*athome+a10[j]*spring+a11[j]*summer+a12[j]*fall+a13[j]*winter-noDWeq[j]*noDW-
noCWeq[j]*noCW)*(1-0.621*senior)*(1+0.3625*nopay);
                                     # Hot Water usage Level at each hr of time schedule

#############
# Variables #
#############

var DECISIONS {i in ADEVICES,j in SCHEDULES} binary ,  <= if ((i = 'AC')             and ( j
< ACEarlTime          or j > ACLateTime))
                                                          or ((i = 'Heating')        and ( j
< ACEarlTime          or j > ACLateTime))
                                                          or ((i = 'Waterheater')    and ( j
< WaterheaterEarlTime or j > WaterheaterLateTime))
                                                          or ((i = 'Fridge')         and ( j
< FridgeEarlTime      or j > FridgeLateTime))
                                                          or ((i = 'Stove')          and ( j
< StoveEarlTime       or j > StoveLateTime))
                                                          or ((i = 'Poolpump')       and ( j
< PoolpumpEarlTime    or j > PoolpumpLateTime))
                                                          or ((i = 'ESD' )           and ( j
< ESDEarlTime         or j > ESDLateTime))
                                                          or ((i = 'Lighting')       and ( j
< LightEarlTime       or j > LightLateTime))
                                                        then 0 else 1;

var TempFrig       {j in SCHEDULES} >= if j = 1 then TempFrigIC        else TempFrigDN
,<= if j = 1 then TempFrigIC        else TempFrigUP; # Fridge Temp at each time schedule
var TempIN         {j in SCHEDULES} >= if j = 1 then TempIC            else TempINDN
,<= if j = 1 then TempIC            else TempINUP; # AC(Indoor) Temperature at each time
schedule
```

```
var TempWaterheater{j in SCHEDULES} >= if j = 1 then TempWaterheaterIC else TempWaterheaterDN
,<= if j = 1 then TempWaterheaterIC else TempWaterheaterUP;# Water Temperature at each hr of
time schedule
var StorESD        {j in SCHEDULES} >= if j = 1 then StorESD0          else ESDMinStorage
,<= if j = 1 then StorESD0          else ESDMaxStorage ; # storage level of ESD at each time
interval in "kwh"

var UStove{SCHEDULES} binary;                # Start Up of Stove at time t
var VStove{SCHEDULES} binary;                # Shutdown of Stove at time t

var Illum  {SCHEDULES} integer >=1  <=6;        # illumenation generated from electric
lighting


###############
# B-Variables #
###############

var BDEC {i in BDEVICES, j in BSCH} binary,   <= if ((i = 'Dishwasher') and ( j <
DishwasherEarlTime or j > DishwasherLateTime))
                                                 or ((i = 'Washer'    ) and ( j <
WasherEarlTime     or j > WasherLateTime    ))
                                                 or ((i = 'Dryer'     ) and ( j <
DryerEarlTime      or j > DryerLateTime     ))
                                                 then 0 else 1 ;

var StorLevDW    {j in BSCH} >= if j = 1 then StorLevDW0     else 0 ,<= if j = 1 then
StorLevDW0 else 1E99; # Dishwasher storage at each hr of time schedule in "kwh"
var StorLevWasher{j in BSCH} >= if j = 1 then StorLevWasher0 else 0 ,<= if j = 1 then
StorLevWasher0 else 1E99;     # Washer storage at each hr of time schedule in "kwh"
var StorLevDryer {j in BSCH} >= if j = 1 then StorLevDryer0  else 0 ,<= if j = 1 then
StorLevDryer0 else 1E99;

var UDishwasher{BSCH} binary;                # Start Up of DW at time t
var VDishwasher{BSCH} binary;                # SHUTDOWN Up of DW at time t
var UWasher{BSCH} binary;                    # Start Up of Washer at time t
var VWasher{BSCH} binary;                    # SHUTDOWN of Washer at time t

var M1    {BSCH};                            # Continus Aux. variable for operation of
dishwasher at each hr of schedule
var M2    {BSCH};                            # Continus Aux. variable for  non-operation of
dishwasher at each hr of schedule
var Z     {BSCH}binary;                              # Binary Aux. variable for dishwasher
operation each hr of time schedule

var M1W   {BSCH};                            # Continus Aux. variable for operation of washer
at each hr of schedule
var M2W  {BSCH};                             # Continus Aux. variable for  non-operation of
washer at each hr of schedule
var ZW    {BSCH}binary;                              # Binary Aux. variable for washer
operation each hr of time schedule

#########
# Model #
#########

minimize TOTAL_PRICE: sum {j in SCHEDULES} (      DECISIONS['AC'         ,j]  *POWER['AC']
*PRICE[j]
                                                + DECISIONS['Heating'    ,j]
*POWER['Heating']      *PRICE[j]
                                                + DECISIONS['Fridge'     ,j]
*POWER['Fridge']       *PRICE[j]
                                                + DECISIONS['Waterheater',j]
*POWER['Waterheater']  *PRICE[j]
                                                + DECISIONS['Poolpump'   ,j]
*POWER['Poolpump']     *PRICE[j]
```

```
                                                + DECISIONS['Stove'      ,j]  *POWER['Stove']
*PRICE[j]
                                                - DECISIONS['ESD'        ,j]  *POWER['ESD']
*PRICE[j]
                                           )

                    + sum {j in BSCH}      (      BDEC['Dryer'           ,j]  *POWER['Dryer']
*PRICE[j]
                                                + BDEC['Washer'          ,j]
*POWER['Washer']       *PRICE[j]
                                                + BDEC['Dishwasher'      ,j]
*POWER['Dishwasher']   *PRICE[j]
                                           )
                                           ;

############# Lighting ###################


       Const_Lighting1 {i in ADEVICES, j in SCHEDULES : i = 'Lighting' and (j >=
LightEarlTime and j <= LightLateTime) }:
               OutIllum[j] + Illum[j] >= ( 1 + ( 1.898 - 0.2041 * PRICE[j] ) ) *ReqIllum[j] ;


############# AC/Heating ########################


       Const_AC1 {j in SCHEDULES, i in ADEVICES: (i = 'AC' or i='Heating') and (j !=1 ) }:

       - TempIN[j]
       + TempIN[j-1] + BetaAC * ActLevH[j] - DECISIONS['AC',j] * AlfaAC +
DECISIONS['Heating', j] * AlfaHeating
        + (OutTemp[j-1] - TempIN[j-1] ) * LambAC = 0;

       Const_AC2 {j in SCHEDULES, i in ADEVICES: (i = 'AC' or i='Heating') }:
               DECISIONS['AC',j] + DECISIONS['Heating',j] <= 1;


        Const_AC3 {j in SCHEDULES, i in ADEVICES: (i = 'AC') }:
               DECISIONS['AC',j] = 0 ;      # in winter cooling should be OFF

/*
        Const_AC3 {j in SCHEDULES, i in ADEVICES: (i = 'Heating') }:
               DECISIONS['Heating',j] = 0;  # in summer heating should be OFF
*/

############# Waterheater ##################


       Const_Waterheater1 {j in SCHEDULES, i in ADEVICES: i = 'Waterheater' and j!=1 }:

       - TempWaterheater[j]
       + TempWaterheater[j-1] - BetaWaterheater * HWL[j] + DECISIONS[i,j] * AlfaWaterheater -
GamaWaterheater = 0;


############# Fridge ######################

       Const_Fridge1 {j in SCHEDULES, i in ADEVICES: i = 'Fridge' and j!=1 }:
       - TempFrig[j] + TempFrig[j-1] + BetaFrig * ActLevF[j] - DECISIONS[i,j] * AlfaFrig +
GamaFrig = 0;


############# Stove #######################

       Const_Stove1 {i in ADEVICES: i = 'Stove'} :         # total operation of Stove is
StovROT
```

```
               sum {j in SCHEDULES} DECISIONS[i,j] = StovROT;

       Const_Stove2 {i in ADEVICES,j in SCHEDULES : i = 'Stove' and  (j >= StoveEarlTime-1 )
and (j <= StoveLateTime+1) } :
               UStove[j] - VStove[j] = DECISIONS[i,j] -DECISIONS[i,j-1] ;

        Const_Stove3 {i in ADEVICES,j in SCHEDULES : i = 'Stove' and  (j >= StoveEarlTime-1 )
and (j <= StoveLateTime+1) } :
               UStove[j] + VStove[j] <= 1 ;

       Const_Stove4 {i in ADEVICES,j in SCHEDULES : i = 'Stove' and (j >= StoveEarlTime-
1+StoveMinUP) and (j <= StoveLateTime+1) } :
               sum {k in j-StoveMinUP+1..j-1} UStove[k] <= DECISIONS[i,j]; # Min. Up time is
StoveMinUP

       Const_Stove5 {i in ADEVICES,j in SCHEDULES : i = 'Stove' and (j >= StoveEarlTime and j
<= StoveLateTime - StovMSOT)} :
               sum {k in j..j+StovMSOT} DECISIONS[i,k] <= StovMSOT ;               # Max. Up
time is StovMSOT

############# Poolpump ######################


       Const_Poolpump1 {i in ADEVICES: i = 'Poolpump'} :                  # operation hrs of
Poolpump is "PoolpumpROT" hrs
               sum {j in SCHEDULES} DECISIONS[i,j] = PoolpumpROT;

########### Energy Storage Device ###########


        Const_ESD1 {j in SCHEDULES, i in ADEVICES: i = 'ESD' and j!=1 }:
               - StorESD[j] + StorESD[j-1] - DECISIONS[i,j]* POWER[i] + ESDChargeRate[j] = 0;


############# B-Dishwasher #################
#      Const_Dishwasher5 {j in BSCH, i in BDEVICES: i = 'Dishwasher' and (j >=
DishwasherEarlTime and j <= DishwasherLateTime) }:
#              if StorLevDW[j] < CapDishwasher
#                      then BDEC[i,j] = 0 ;
###########


       Const_Dishwasher1 {j in BSCH, i in BDEVICES: i = 'Dishwasher' and j!=1 }:
               - StorLevDW[j]
               + StorLevDW[j-1]  - BDEC[i,j-1]* CapDishwasher/DishwasherMinUP +
BetaDishwasher * BActLevD[j] = 0;


        Const_Dishwasher2 {i in BDEVICES,j in BSCH : i = 'Dishwasher' and (j >=
DishwasherEarlTime-1)and (j <= DishwasherLateTime+1)  } :
               UDishwasher[j]- VDishwasher[j] = BDEC[i,j] - BDEC[i,j-1] ;

        Const_Dishwasher3 {i in BDEVICES,j in BSCH : i = 'Dishwasher' and (j >=
DishwasherEarlTime-1 )and (j <= DishwasherLateTime+1)  } :
               UDishwasher[j] + VDishwasher[j] <= 1 ;


        Const_Dishwasher4 {i in BDEVICES,j in BSCH : i = 'Dishwasher' and (j
>=DishwasherEarlTime + DishwasherMinUP-1) and (j <= DishwasherLateTime+1)} :
               sum {k in j-DishwasherMinUP+1..j-1} UDishwasher[k] <= BDEC[i,j];   # Min. Up
time is DishwasherMinUP

       Const_Dishwasher5 {i in BDEVICES,j in BSCH : i = 'Dishwasher' and (j
>=DishwasherEarlTime + DishwasherMinDN-1 ) and (j <= DishwasherLateTime+1)} :
               sum {k in j-DishwasherMinDN+1..j-1} VDishwasher[k] <= 1 - BDEC[i,j];    # Min.
Down time is DishwasherMinDN
```

```
        Const_DishWasher6 {j in BSCH, i in BDEVICES: i = 'Dishwasher' and j=25 }:

               StorLevDW[j] <= 0.9 * CapDishwasher ;

############# B-Washer ######################


       Const_Washer1 {j in BSCH, i in BDEVICES: i = 'Washer' and j!=1 }:
               - StorLevWasher[j]
               + StorLevWasher[j-1] - BDEC[i,j-1] * CapWasher/WasherMinUP +  person * 0.25 =
0;


        Const_Washer2 {i in BDEVICES,j in BSCH : i = 'Washer' and (j >= WasherEarlTime-1 )
and (j <= WasherLateTime+1) } :
               UWasher[j] - VWasher[j] = BDEC[i,j] - BDEC[i,j-1] ;

        Const_Washer3 {i in BDEVICES,j in BSCH : i = 'Washer' and (j >= WasherEarlTime-1 )
and (j <= WasherLateTime+1) } :
               UWasher[j] + VWasher[j] <= 1 ;


       Const_Washer4 {i in BDEVICES,j in BSCH : i = 'Washer' and (j >=WasherEarlTime +
WasherMinUP-1) and (j <= WasherLateTime+1)} :
               sum {k in j-WasherMinUP+1..j-1} UWasher[k] <= BDEC[i,j];    # Min. Up time is
WasherMinUP

       Const_Washer5 {i in BDEVICES,j in BSCH : i = 'Washer' and (j >=WasherEarlTime +
WasherMinDN-1) and (j <= WasherLateTime+1)} :
               sum {k in j-WasherMinDN+1..j-1} VWasher[k] <= 1 - BDEC[i,j] ;# Min. Down time
is WasherMinDN

       Const_Washer6 {j in BSCH, i in BDEVICES: i = 'Washer' and j=25 }:
               StorLevWasher[j] <= 0.9 * CapWasher;

############# B-Dryer ######################


       Const_Dryer1 {j in BSCH, i in BDEVICES: i = 'Dryer' and j!=1 }:
               - StorLevDryer[j]
               + StorLevDryer[j-1] - BDEC[i ,j-1] * CapDryer/WasherMinUP +  BDEC['Washer',j-
1] * CapWasher/WasherMinUP = 0;


######## B- Washer/Dryer Coordination #########

# These constraints garantee that Dryer operates after Washer, and in a time less than
MTGapWasherDryer of operation of Washer

       Const_WaherDryer1 {j in BSCH, i in BDEVICES: (j >= DryerEarlTime and j <=
DryerLateTime) and (i = 'Dryer' or i='Washer') }:
               BDEC['Dryer',j] <=  sum {k in 1..MTGapWashDryer} BDEC['Washer',j-k];

       Const_WaherDryer2 {j in BSCH, i in BDEVICES: (i = 'Dryer' or i='Washer') and (j >=
DryerEarlTime-1) and (j <= DryerLateTime+1)}:
               BDEC['Dryer',j] + BDEC['Washer',j] <= 1;

        Const_Dryer3 {j in BSCH, i in BDEVICES: i = 'Dryer' and j=23 }:
               StorLevDryer[j] <= 0;


#########
solve;

#########
# Print #
#########
```

```
printf " \n"   ;


printf "TOTAL_PRICE= %6.2f " ,
         sum {j in SCHEDULES} (   DECISIONS['AC'         ,j]  *POWER['AC']
*PRICE[j]
                                + DECISIONS['Heating'    ,j]  *POWER['Heating']
*PRICE[j]
                                + DECISIONS['Fridge'     ,j]  *POWER['Fridge']
*PRICE[j]
                                + DECISIONS['Waterheater',j]  *POWER['Waterheater']
*PRICE[j]
                                + DECISIONS['Stove'      ,j]  *POWER['Stove']
*PRICE[j]
                                + DECISIONS['Poolpump'   ,j]  *POWER['Poolpump']
*PRICE[j]
                                + Illum[j]                    *POWER['Lighting']
*PRICE[j]
                                - DECISIONS['ESD'        ,j]  *POWER['ESD']
*PRICE[j]

                              ) / 400000

       + sum {j in BSCH}    (     BDEC['Dryer'           ,j]  *POWER['Dryer']        *PRICE[j]
                                + BDEC['Washer'           ,j]  *POWER['Washer']
*PRICE[j]
                                + BDEC['Dishwasher'      ,j]  *POWER['Dishwasher']
*PRICE[j]
                              ) / 100000 ;



printf "TOTAL_consumption(KWH)= %6.2f " ,
          sum {j in SCHEDULES} (   DECISIONS['AC'         ,j]  *POWER['AC']
                                 + DECISIONS['Heating'    ,j]  *POWER['Heating']
                                 + DECISIONS['Fridge'     ,j]  *POWER['Fridge']
                                 + DECISIONS['Waterheater',j]  *POWER['Waterheater']
                                 + DECISIONS['Stove'      ,j]  *POWER['Stove']
                                 + DECISIONS['Poolpump'   ,j]  *POWER['Poolpump']
                                 + Illum[j]                    *POWER['Lighting']
                                 - DECISIONS['ESD'        ,j]  *POWER['ESD']

                               ) / 4000

       + sum {j in BSCH}      (   BDEC['Dryer'           ,j]  *POWER['Dryer']
                                 + BDEC['Washer'           ,j]  *POWER['Washer']
                                 + BDEC['Dishwasher'      ,j]  *POWER['Dishwasher']
                               ) / 1000 ;


printf "\n"    ;
printf "\n"    ;
printf "Heating_consumption(KWH)= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i =
'Heating'} ( DECISIONS[i,j]*POWER[i]) / 4000 ;
printf "Heating_ON= %3d ", sum {j in SCHEDULES , i in ADEVICES : i = 'Heating'}
DECISIONS[i,j] ;
printf "Heating_price= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'Heating'} (
DECISIONS[i,j]*POWER[i]*PRICE[j]) / 4000 ;
printf "\n"    ;
printf "\n"    ;
printf "cooling_consumption(KWH)= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'AC'} (
DECISIONS[i,j]*POWER[i]) / 4000 ;
printf "cooling_ON= %3d ", sum {j in SCHEDULES , i in ADEVICES : i = 'AC'} DECISIONS[i,j] ;
printf "cooling_price= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'AC'} (
DECISIONS[i,j]*POWER[i]*PRICE[j]) / 4000 ;
printf "\n"    ;
```

```
printf "\n"    ;
printf "Waterheater_consumption(KWH)= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i =
'Waterheater'} ( DECISIONS[i,j]*POWER[i]) / 4000 ;
printf "Waterheater_ON= %3d ", sum {j in SCHEDULES , i in ADEVICES : i = 'Waterheater'}
DECISIONS[i,j] ;
printf "Waterheater_price= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'Waterheater'}
( DECISIONS[i,j]*POWER[i]*PRICE[j]) / 4000 ;
printf "hotwater_use(liters)=  %6.2f ", sum {j in SCHEDULES } ( HWL[j]/4) ;
printf "\n"    ;
printf "\n"    ;
printf "Fridge_consumption(KWH)= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'Fridge'}
( DECISIONS[i,j]*POWER[i]) / 4000 ;
printf "Fridge_ON= %3d ", sum {j in SCHEDULES , i in ADEVICES : i = 'Fridge'} DECISIONS[i,j]
;
printf "Fridge_price= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'Fridge'} (
DECISIONS[i,j]*POWER[i]*PRICE[j]) / 4000 ;
printf "\n"    ;
printf "\n"    ;
printf "POOLPUMP_consumption(KWH)= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i =
'Poolpump'} ( DECISIONS[i,j]*POWER[i]) / 4000 ;
printf "POOLPUMP_ON= %3d ", sum {j in SCHEDULES , i in ADEVICES : i = 'Poolpump'}
DECISIONS[i,j] ;
printf "POOLPUMP_price= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'Poolpump'} (
DECISIONS[i,j]*POWER[i]*PRICE[j]) / 4000 ;
printf "\n"    ;
printf "\n"    ;
printf "stove_consumption(KWH)= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'Stove'} (
DECISIONS[i,j]*POWER[i]) / 4000 ;
printf "stove_ON= %3d ", sum {j in SCHEDULES , i in ADEVICES : i = 'Stove'} DECISIONS[i,j] ;
printf "stove_price= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'Stove'} (
DECISIONS[i,j]*POWER[i]*PRICE[j]) / 4000 ;
printf "\n"    ;
printf "\n"    ;
printf "Lighting_consumption(KWH)= %6.2f ", (sum {j in SCHEDULES , i in ADEVICES : i =
'Lighting'} ( Illum[j]*POWER[i]) / 4000);
printf "Lighting_price= %6.2f ", (sum {j in SCHEDULES , i in ADEVICES : i = 'Lighting'} (
Illum[j]*POWER[i]*PRICE[j]) / 4000) ;
printf "\n"    ;
printf "\n"    ;
printf "ESD_injection(KWH)= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'ESD'} (
DECISIONS[i,j]*POWER[i]) / 4000 ;
printf "ESD_saving= %6.2f ", sum {j in SCHEDULES , i in ADEVICES : i = 'ESD'} (
DECISIONS[i,j]*POWER[i]*PRICE[j]) / 4000 ;
printf "\n" ;


printf "------------------------------------------------------------\n"    ;
printf "| Time|           AC  Watheat Fridge Stove Heating Poolpump ESD    |\n"    ;
printf "------------------------------------------------------------|\n"   ;



for{j in SCHEDULES} {
                      printf "%1s %3d %1s","|", j, "|" ;
                      for {i in ADEVICES: i != 'Lighting'}  printf "%7d" , DECISIONS[i,j] ;
                        printf "%5s \n", "    |"   ;

                   }

printf "------------------------------------------------------------\n"    ;



printf "-----------------------------------------------------------------------------\n"  ;

printf "Time   T_out     T_WH      T_Fr    Tempin    HWL      Act-LevH Act-LevF PRICE\n";
```

```
for{j in SCHEDULES} {
               printf "%3d %8.2f %8.2f %8.2f %8.2f %8.2f  %8.2f %8.2f %8.2f\n", j
,OutTemp[j], TempWaterheater[j], TempFrig[j],TempIN[j],HWL[j],ActLevH[j],ActLevF[j], PRICE[j]
                      ;
                   }

printf "-----------------------------------------------------------------------------\n"  ;

printf "-----------------------------------------------------------------------------------
\n"    ;

printf " Time PRICE Dishwash StorLev-DW clothwash StorLev-wash Dryer StoreLev-Dryer
ActLevD\n";

for{j in BSCH} {
               printf " %3d  %4.1f   %4d  %8d  %8d %10d  %8d  %8d %12.2f\n", j , BPRICE[j],
BDEC['Dishwasher',j], StorLevDW[j],BDEC['Washer',j], StorLevWasher[j] ,BDEC['Dryer',j],
StorLevDryer[j],BActLevD[j]
                      ;
                   }

printf "-----------------------------------------------------------------------------------
\n"    ;

printf "\n"    ;
printf "DishWash_PRICE($)= %6.2f ", sum {j in BSCH , i in BDEVICES : i = 'Dishwasher'} (
BDEC[i,j]*POWER[i]*BPRICE[j]) / 100000 ;
printf "\n"    ;
printf "clothhWash_PRICE($)= %6.2f ", sum {j in BSCH , i in BDEVICES : i = 'Washer'} (
BDEC[i,j]*POWER[i]*BPRICE[j]) / 100000 ;
printf "\n"    ;
printf "Dryer_PRICE($)= %6.2f ", sum {j in BSCH , i in BDEVICES : i = 'Dryer'} (
BDEC[i,j]*POWER[i]*BPRICE[j]) / 100000 ;
printf "\n"    ;

printf "-----------------------------------------------------------------------------\n"  ;


printf "--------------------------------------------------------------\n" ;

printf "Time  OUTiLLum  req-illum  decision-L  illum  \n";

for{j in SCHEDULES} {
               printf " %3d %8.2f %10.2f %10.2f %10.2f
\n",j,OutIllum[j],ReqIllum[j],DECISIONS['Lighting',j],Illum[j]
                      ;
                   }

printf "-------------------------------------------------------------\n";

printf "-----------------------------------------------\n"  ;

printf "Time   ESDstorage   ESD   ESDchargerate   PRICE\n";

for{j in SCHEDULES} {
               printf "%3d %12.2f %6d %14.2f %8.2f\n", j ,
StorESD[j],DECISIONS['ESD',j],ESDChargeRate[j], PRICE[j]
                      ;
                   }

printf "-----------------------------------------------\n"  ;
```

```
#Data file:
set ADEVICES:= AC Waterheater Fridge Stove Heating Poolpump ESD Lighting;
set BDEVICES:= Dishwasher Washer Dryer;
param POWER :=
       AC             11500
       Waterheater    3600
       Fridge         250
       Stove          1500
       Heating        400
       Poolpump       750
       ESD            100
       Lighting       150
       Dishwasher     600
       Washer         450
       Dryer          1100
;

param StorLevDW0         :=  10  ;
param BetaDishwasher     :=  1.5 ;
param CapDishwasher      :=  100 ;
param DishwasherEarlTime :=  7   ;
param DishwasherLateTime :=  23  ;
param DishwasherMinUP    :=  2   ;
param DishwasherMinDN    :=  1   ;

param StorLevWasher0     :=  85  ;
param CapWasher          :=  100 ;
param WasherEarlTime     :=  7   ;
param WasherLateTime     :=  23  ;
param WasherMinUP        :=  2   ;
param WasherMinDN        :=  1   ;
param person             :=  4   ;

param StorLevDryer0      :=  00  ;
param CapDryer           :=  100 ;
param DryerEarlTime      :=  7   ;
param DryerLateTime      :=  24  ;

param MTGapWashDryer     :=  3   ;


param:
BSCH:          BEEC   BPRICE:=

1              150    4.4
2              100    4.4
3              87.5   4.4
4              125    4.4
5              75     4.4
6              70     4.4
7              100    4.4
8              262.5  9.3
9              150    9.3
10             200    9.3
11             350    9.3
12             800    8
13             1100   8
14             150    8
15             100    8
16             125    8
17             225    8
18             187.5  9.3
19             562.5  9.3
20             187.5  9.3
21             212.5  9.3
22             250    4.4
23             75     4.4
```

```
24              75      4.4
25              75      4.4
;

param TempFrigUP           :=  8     ;
param TempFrigDN           :=  2     ;
param TempFrigIC           :=  5     ;
param BetaFrig             :=  0.28  ;
param AlfaFrig             :=  5.5   ;
param GamaFrig             :=  1.21  ;
param FridgeEarlTime       :=  1     ;
param FridgeLateTime       :=  96    ;


param TempWaterheaterUP    :=  65    ;
param TempWaterheaterDN    :=  55    ;
param TempWaterheaterIC    :=  60    ;
param BetaWaterheater      :=  0.068 ;
param AlfaWaterheater      :=  4.44  ;
param GamaWaterheater      :=  0.083 ;
param WaterheaterEarlTime  :=  1     ;
param WaterheaterLateTime  :=  96    ;


param StoveMinUP           :=  4     ;
param StovMSOT             :=  16    ;
param StovROT              :=  24    ;

param TempINUP             :=  23    ;
param TempINDN             :=  17    ;
param TempIC               :=  20    ;
param AlfaHeating          :=  1.02  ;
param AlfaAC               :=  0.51  ;
param BetaAC               :=  0.0275;
param LambAC               :=  0.0075;
param ACEarlTime           :=  1     ;
param ACLateTime           :=  96    ;


param PoolpumpROT          :=  32    ;
param PoolpumpEarlTime     :=  1     ;
param PoolpumpLateTime     :=  96    ;


param ESDEarlTime          :=  2     ;
param ESDLateTime          :=  96    ;
param ESDMinStorage        :=  250   ;
param ESDMaxStorage        :=  2000  ;
param StorESD0             :=  250   ;


param   StoveEarlTime     :=  20    ;
param   StoveLateTime     :=  96    ;
param   LightEarlTime     :=  1     ;
param   LightLateTime     :=  96    ;

param  per      := 4     ;    # number of persons in a house
param  age1     := 1     ;    # number of preschool children (0-5 yrs)
param  age2     := 1     ;    # number of school age children (6-13 yrs)
param  age3     := 2     ;    # number of adults (14 yrs and over)
param  therm    := 60    ;    # water heater thermostat setting, (°C)
param  tankz    := 185   ;    # water heater nominal tank size, (Liters)
param  wtmp     := 8     ;    # water heater inlet water temperature, (°C)
param  atmp     := -10   ;    # outdoor air ambient temperature,average on each day (°C)
param  athome   := 1     ;    # a dummy parameter for the presence of adults at home during
day (unemployed person)
param  spring   := 0     ;    # a dummy parameter for Spring (1 if “spring”, zero otherwise)
```

```
param  summer   := 0     ;    # a dummy parameter for Summer (1 if “spring”, zero otherwise)
param  fall     := 0     ;    # a dummy parameter for fall (1 if “spring”, zero otherwise)
param  winter   := 1     ;    # a dummy parameter for winter (1 if “spring”, zero otherwise)
param  noDW     := 0     ;    # a dummy parameter for dishwasher (1 if dishwasher is not
available, zero otherwise)
param  noCW     := 0     ;    # a dummy parameter for clothes washer (1 if washer is not
available, zero otherwise)
param  senior   := 0     ;    # a dummy parameter for senior = 1 if this is a senior-only
household, = 0 otherwise
param  nopay    := 0     ;    # a dummy parameter = 1 if household does not pay for hot
water, = 0 otherwise



param:
SCHEDULES:     EEC     ESDChargeRate  OutTemp  ReqIllum      OutIllum       PRICE   a0      a1
       a2      a3      a4      a5      a6      a7      a8      a9      a10     a11     a12
       a13:=



1      150     55      -17     0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
2      150     55      -17     0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
3      150     55      -17     0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
4      150     55      -17     0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
5      100     55      -14.3   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
6      100     55      -14.3   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
7      100     55      -14.3   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
8      100     55      -14.3   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
9      87.5    55      -15.6   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
10     87.5    55      -15.6   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
11     87.5    55      -15.6   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
12     87.5    55      -15.6   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
13     125     55      -16.5   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
14     125     55      -16.5   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
15     125     55      -16.5   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
16     125     55      -16.5   0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
17     75      55      -19     0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
18     75      55      -19     0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
19     75      55      -19     0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
20     75      55      -19     0.1     0       4.4     0       0.6163  0       0       0       0
       -0.0017 0       0       0       0       0       0       0.5523
21     70      55      -20.1   1       0.1     4.4     2.0956  0       0       3.483   7.9861  0
       0.0269  -0.5424 -0.6603 -3.6609 0       13.601  0       0
22     70      55      -20.1   1       0.1     4.4     2.0956  0       0       3.483   7.9861  0
       0.0269  -0.5424 -0.6603 -3.6609 0       13.601  0       0
23     70      55      -20.1   1       0.1     4.4     2.0956  0       0       3.483   7.9861  0
       0.0269  -0.5424 -0.6603 -3.6609 0       13.601  0       0
```

```
24      70      55      -20.1   1       0.1     4.4     2.0956  0       0       3.483   7.9861  0
        0.0269  -0.5424 -0.6603 -3.6609 0               13.601  0       0
25      100     60      -19.6   1       0.1     4.4     2.0956  0       0       3.483   7.9861  0
        0.0269  -0.5424 -0.6603 -3.6609 0               13.601  0       0
26      100     60      -19.6   1       0.1     4.4     2.0956  0       0       3.483   7.9861  0
        0.0269  -0.5424 -0.6603 -3.6609 0               13.601  0       0
27      100     60      -19.6   1       0.1     4.4     2.0956  0       0       3.483   7.9861  0
        0.0269  -0.5424 -0.6603 -3.6609 0               13.601  0       0
28      100     60      -19.6   1       0.1     4.4     2.0956  0       0       3.483   7.9861  0
        0.0269  -0.5424 -0.6603 -3.6609 0               13.601  0       0
29      262.5   60      -23.3   1       0.1     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
30      262.5   60      -23.3   1       0.1     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
31      262.5   60      -23.3   1       0.1     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
32      262.5   60      -23.3   1       0.1     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
33      150     55      -21.5   2       0.5     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
34      150     55      -21.5   2       0.5     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
35      150     55      -21.5   2       0.5     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
36      150     55      -21.5   2       0.5     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
37      200     55      -21     2       0.5     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
38      200     55      -21     2       0.5     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
39      200     55      -21     2       0.5     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
40      200     55      -21     2       0.5     9.3     0       0       1.0853  1.5331  2.4972  0
        0       0       0       9.0418  0       0       -1.6353 2.1403
41      350     55      -16.9   2       1       9.3     -0.3876 0       0.9668  1.0849  2.0956  -
0.0218  0       0       0       6.1986  0       0       -1.6834 1.5187
42      350     55      -16.9   2       1       9.3     -0.3876 0       0.9668  1.0849  2.0956  -
0.0218  0       0       0       6.1986  0       0       -1.6834 1.5187
43      350     55      -16.9   2       1       9.3     -0.3876 0       0.9668  1.0849  2.0956  -
0.0218  0       0       0       6.1986  0       0       -1.6834 1.5187
44      350     55      -16.9   2       1       9.3     -0.3876 0       0.9668  1.0849  2.0956  -
0.0218  0       0       0       6.1986  0       0       -1.6834 1.5187
45      800     55      -14     2       1       8       -0.3876 0       0.9668  1.0849  2.0956  -
0.0218  0       0       0       6.1986  0       0       -1.6834 1.5187
46      800     55      -14     2       1       8       -0.3876 0       0.9668  1.0849  2.0956  -
0.0218  0       0       0       6.1986  0       0       -1.6834 1.5187
47      800     55      -14     2       1       8       -0.3876 0       0.9668  1.0849  2.0956  -
0.0218  0       0       0       6.1986  0       0       -1.6834 1.5187
48      800     55      -14     2       1       8       -0.3876 0       0.9668  1.0849  2.0956  -
0.0218  0       0       0       6.1986  0       0       -1.6834 1.5187
49      1100    60      -13     2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
50      1100    60      -13     2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
51      1100    60      -13     2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
52      1100    60      -13     2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
53      150     60      -12.8   2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
54      150     60      -12.8   2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
55      150     60      -12.8   2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
56      150     60      -12.8   2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
```

```
57      100     60      -13.1   2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
58      100     60      -13.1   2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
59      100     60      -13.1   2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
60      100     60      -13.1   2       1       8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
61      125     60      -13.2   2       0.5     8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
62      125     60      -13.2   2       0.5     8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
63      125     60      -13.2   2       0.5     8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
64      125     60      -13.2   2       0.5     8       -0.2907 0       1.979   1.2     2.3072  -
0.0906  0.0083  0       -0.0743 4.0228  0       0       0       2.5854
65      225     55      -13.5   3       0.5     8       0.7753  0       1.5679  2.0415  3.6018  0
        0       -0.3134 -0.357  5.3492  0       3.6855  0       3.656
66      225     55      -13.5   3       0.5     8       0.7753  0       1.5679  2.0415  3.6018  0
        0       -0.3134 -0.357  5.3492  0       3.6855  0       3.656
67      225     55      -13.5   3       0.5     8       0.7753  0       1.5679  2.0415  3.6018  0
        0       -0.3134 -0.357  5.3492  0       3.6855  0       3.656
68      225     55      -13.5   3       0.5     8       0.7753  0       1.5679  2.0415  3.6018  0
        0       -0.3134 -0.357  5.3492  0       3.6855  0       3.656
69      187.5   55      -14.4   3       0.1     9.3     0.7753  0       1.5679  2.0415  3.6018  0
        0       -0.3134 -0.357  5.3492  0       3.6855  0       3.656
70      187.5   55      -14.4   3       0.1     9.3     0.7753  0       1.5679  2.0415  3.6018  0
        0       -0.3134 -0.357  5.3492  0       3.6855  0       3.656
71      187.5   55      -14.4   3       0.1     9.3     0.7753  0       1.5679  2.0415  3.6018  0
        0       -0.3134 -0.357  5.3492  0       3.6855  0       3.656
72      187.5   55      -14.4   3       0.1     9.3     0.7753  0       1.5679  2.0415  3.6018  0
        0       -0.3134 -0.357  5.3492  0       3.6855  0       3.656
73      562.5   55      -15.2   3       0.1     9.3     4.4577  0       2.7456  4.4092  3.3455  -
0.1015  0.0187  0       -0.3523 0       0       8.0527  -3.2509 0
74      562.5   55      -15.2   3       0.1     9.3     4.4577  0       2.7456  4.4092  3.3455  -
0.1015  0.0187  0       -0.3523 0       0       8.0527  -3.2509 0
75      562.5   55      -15.2   3       0.1     9.3     4.4577  0       2.7456  4.4092  3.3455  -
0.1015  0.0187  0       -0.3523 0       0       8.0527  -3.2509 0
76      562.5   60      -15.2   3       0.1     9.3     4.4577  0       2.7456  4.4092  3.3455  -
0.1015  0.0187  0       -0.3523 0       0       8.0527  -3.2509 0
77      187.5   60      -15.3   3       0       9.3     4.4577  0       2.7456  4.4092  3.3455  -
0.1015  0.0187  0       -0.3523 0       0       8.0527  -3.2509 0
78      187.5   60      -15.3   3       0       9.3     4.4577  0       2.7456  4.4092  3.3455  -
0.1015  0.0187  0       -0.3523 0       0       8.0527  -3.2509 0
79      187.5   60      -15.3   2       0       9.3     4.4577  0       2.7456  4.4092  3.3455  -
0.1015  0.0187  0       -0.3523 0       0       8.0527  -3.2509 0
80      187.5   60      -15.3   2       0       9.3     4.4577  0       2.7456  4.4092  3.3455  -
0.1015  0.0187  0       -0.3523 0       0       8.0527  -3.2509 0
81      212.5   60      -15.7   2       0       9.3     4.2881  0       1.4434  3.4394  2.5135  -
0.0436  0       0       -0.2848 -2.4166 0       3.3773  -3.5511 0
82      212.5   60      -15.7   2       0       9.3     4.2881  0       1.4434  3.4394  2.5135  -
0.0436  0       0       -0.2848 -2.4166 0       3.3773  -3.5511 0
83      212.5   60      -15.7   2       0       9.3     4.2881  0       1.4434  3.4394  2.5135  -
0.0436  0       0       -0.2848 -2.4166 0       3.3773  -3.5511 0
84      212.5   60      -15.7   2       0       9.3     4.2881  0       1.4434  3.4394  2.5135  -
0.0436  0       0       -0.2848 -2.4166 0       3.3773  -3.5511 0
85      250     60      -15.9   2       0       4.4     4.2881  0       1.4434  3.4394  2.5135  -
0.0436  0       0       -0.2848 -2.4166 0       3.3773  -3.5511 0
86      250     60      -15.9   2       0       4.4     4.2881  0       1.4434  3.4394  2.5135  -
0.0436  0       0       -0.2848 -2.4166 0       3.3773  -3.5511 0
87      250     60      -15.9   2       0       4.4     4.2881  0       1.4434  3.4394  2.5135  -
0.0436  0       0       -0.2848 -2.4166 0       3.3773  -3.5511 0
88      250     55      -15.9   2       0       4.4     4.2881  0       1.4434  3.4394  2.5135  -
0.0436  0       0       -0.2848 -2.4166 0       3.3773  -3.5511 0
89      75      55      -17     2       0       4.4     0       0.6163  0       0       0       0
        -0.0017 0       0       0       0       0       0       0.5523
```

```
90      75      55      -17     2       0       4.4     0       0.6163  0       0       0       0
        -0.0017 0       0       0       0       0       0       0.5523
91      75      55      -17     2       0       4.4     0       0.6163  0       0       0       0
        -0.0017 0       0       0       0       0       0       0.5523
92      75      55      -17     2       0       4.4     0       0.6163  0       0       0       0
        -0.0017 0       0       0       0       0       0       0.5523
93      75      55      -17.7   1       0       4.4     0       0.6163  0       0       0       0
        -0.0017 0       0       0       0       0       0       0.5523
94      75      55      -17.7   1       0       4.4     0       0.6163  0       0       0       0
        -0.0017 0       0       0       0       0       0       0.5523
95      75      55      -17.7   1       0       4.4     0       0.6163  0       0       0       0
        -0.0017 0       0       0       0       0       0       0.5523
96      75      55      -17.7   1       0       4.4     0       0.6163  0       0       0       0
        -0.0017 0       0       0       0       0       0       0.5523
97      75      55      -17.9   1       0       4.4     0       0.6163  0       0       0       0
        -0.0017 0       0       0       0       0       0       0.5523
;
```

Appendix B

Code CDMRes4B.mod

```
###########################################################################
# By Mohammad Chehreghani Bozchalui @ Nov    2009                         #
# modefied by Hussin Hassen on 21 Feb. 2010                               #
# @ University of Waterloo                                                #
# 96 time schudels for all devices are used here                          #
# DW AND W equations are reduced                                          #
###########################################################################
#reset;
set DEVICES;
set SCHEDULES;
#############
#Parameters #
#############

param PRICE {SCHEDULES};                            # Price per schedule in "cent/kwh"
param POWER {DEVICES};                          # Power cunsumption of each Device in "kw"
param POWERCONS {SCHEDULES};                 # Total power consumption at each time schedule

param EEC {j in SCHEDULES};                     # ELECTRICAL ENERGY CONSUMPTION in KWhr every
15 min. (of the previous week)
param EEC1{j in SCHEDULES}:= EEC[j] less (0.5 * (sum {k in 1..96} EEC[k])/96);#Dummy
parameter for calculating ActLev
param ActLevA {j in SCHEDULES}:= (100*EEC [j]) / sum {k in 1..96} EEC[k] ; # HEATING Activity
Level at each hr of time schedule
param ActLevB {j in SCHEDULES}:= (100*EEC1[j]) / sum {k in 1..96} EEC1[k]; # FRG usage
Activity Level at each hr of time schedule

param OutIllum1 {SCHEDULES};                 # daylight Illumination @ zone 1, which entered
from outside
param ReqIllum1 {SCHEDULES};                 # minimum required Illumination for zone 1.
param OutIllum2 {SCHEDULES};                 # daylight Illumination @ zone 2, which entered
from outside
param ReqIllum2 {SCHEDULES};                 # minimum required Illumination for zone 2.
param OutIllum3 {SCHEDULES};                 # daylight Illumination @ zone 1, which entered
from outside
param ReqIllum3 {SCHEDULES};                 # minimum required Illumination for zone 1.
param OutIllum4 {SCHEDULES};                 # daylight Illumination @ zone 2, which entered
from outside
param ReqIllum4 {SCHEDULES};                 # minimum required Illumination for zone 2.
param OutIllum5 {SCHEDULES};                 # daylight Illumination @ zone 1, which entered
from outside
param ReqIllum5 {SCHEDULES};                 # minimum required Illumination for zone 1.
param OutIllum6 {SCHEDULES};                 # daylight Illumination @ zone 2, which entered
from outside
param ReqIllum6 {SCHEDULES};                 # minimum required Illumination for zone 2.
param Illum1  {j in SCHEDULES}:= if OutIllum1[j] >= ReqIllum1[j]  then 0   else
ReqIllum1[j];# illumenation generated from electric lighting at zone 1.
param Illum2  {j in SCHEDULES}:= if OutIllum2[j] >= ReqIllum2[j]  then 0   else
ReqIllum2[j];# illumenation generated from electric lighting at zone 2.
param Illum3  {j in SCHEDULES}:= if OutIllum3[j] >= ReqIllum3[j]  then 0   else
ReqIllum3[j];# illumenation generated from electric lighting at zone 3.
param Illum4  {j in SCHEDULES}:= if OutIllum4[j] >= ReqIllum4[j]  then 0   else
ReqIllum4[j];# illumenation generated from electric lighting at zone 4.
param Illum5  {j in SCHEDULES}:= if OutIllum5[j] >= ReqIllum5[j]  then 0   else
ReqIllum5[j];# illumenation generated from electric lighting at zone 5.
param Illum6  {j in SCHEDULES}:= if OutIllum6[j] >= ReqIllum6[j]  then 0   else
ReqIllum6[j];# illumenation generated from electric lighting at zone 6.

param TempFrigUP;                            # Fridge Temperature Up limit
```

```
param TempFrigDN;                            # Fridge Temperature Down limit
param TempFrigIC;                            # Fridge Temperature Setting (ideal Temp for
Fridge)
param BetaFrig;                                     # ActLev Coefficient in the TempFrig
formula.
param AlfaFrig;                                     # Cooling effect coefficient of previous
ON time interval in the TempFrig formula
param GamaFrig;                                     # Warming effect coefficient of previous
OFF time interval in the TempFrig formula

param TempWaterheaterUP;                     # Waterheater Temperature Up limit
param TempWaterheaterDN;
param TempWaterheaterIC;
param BetaWaterheater ;
param AlfaWaterheater;
param GamaWaterheater;


param AlfaHeating;
param TempINUP;                                     # Indoor Temperature Up limit
param TempINDN;                                     # Indoor Temperature Down limit
param TempIC;                                 # indoor starting initial temparature
param OutTemp {SCHEDULES};                   # Outdoor Temperature at each hr of time
schedule (measured)
param BetaAC;                                # ActLev Coefficient in the TempIN formula.
param AlfaAC;                                # Cooling effect coefficient of previous ON time
interval in the TempIN formula
param LambAC;                                # represents the effect of outdoor temperature
of previous ime interval in the TempIN formula


param PoolpumpROT;                           # Poolpump Req. Operation Time
param PoolpumpEarlTime;
param PoolpumpLateTime;


param PVEarlTime;
param PVLateTime;
param PVgen{SCHEDULES};                       # PV power generation rate at time interval t,
(kw/ 15min)
param BATMinStorage;
param BATMaxStorage;
param StorBAT0;
param BATCharge {j in SCHEDULES}:= if PVgen[j] >= POWER['CR']  then POWER['CR'] else
PVgen[j];# charging power



param StoveMinUP;
param StovMSOT;
param StovROT;                                # Stov Req. Operation Time

param  LightEarlTime;                         # Earliest Possible Scheduling time for Lighting
(Based on the user preference)
param  LightLateTime;                         # Latest Possible Scheduling time for Lighting
(Based on the user preference)
param  ACEarlTime;
param  ACLateTime;
param  WaterheaterEarlTime;
param  WaterheaterLateTime;
param  FridgeEarlTime;
param  FridgeLateTime;
param  StoveEarlTime;
param  StoveLateTime;
param  TWHEarlTime;
param  TWHLateTime;
```

```
param   DishwasherEarlTime;
param   DishwasherLateTime;
param   DishwasherMinUP;
param   DishwasherMinDN;
param    DWMSOT;

param   WasherEarlTime;
param   WasherLateTime;
param   WasherMinUP;
param   WasherMinDN;
param    WMSOT;

param   DryerEarlTime;
param   DryerLateTime;
param   DryerMinUP;
param   DryerMinDN;

param MTGapWashDryer;                      # Maximum Allowable Time Gap between operations
of Washer and Dryer

#the followings are Hourly Hot water use CALCULATIONS

param  per ;    # number of persons in a house
param  age1;    # number of preschool children (0-5 yrs)
param  age2;    # number of school age children (6-13 yrs)
param  age3;    # number of adults (14 yrs and over)
param  therm;   # water heater thermostat setting, (°C)
param  tankz;   # water heater nominal tank size, (Liters)
param  wtmp;    # water heater inlet water temperature, (°C)
param  atmp;    # outdoor air ambient temperature, (°C)
param  athome;  # a dummy parameter for the presence of adults at home during day (unemployed
person)
param  spring;  # a dummy parameter for Spring (1 if “spring”, zero otherwise)
param  summer;  # a dummy parameter for Summer (1 if “spring”, zero otherwise)
param  fall;    # a dummy parameter for fall (1 if “spring”, zero otherwise)
param  winter;  # a dummy parameter for winter (1 if “spring”, zero otherwise)
param  noDW;    # a dummy parameter (1 if dishwasher is not available, zero otherwise)
param  noCW;    # a dummy parameter (1 if washer is not available, zero otherwise)
param  noDWeq{j in SCHEDULES}:= if (65 <=j and j<= 80) then (0.5-atmp/80)*(2.62*per+5.054*
sqrt(per))/4 else 0; #a dummy function indicating impact of not owning a dish washer
param  noCWeq{j in SCHEDULES}:= if (65 <=j and j<= 80) then (0.5-atmp/80)*(4.4242*per+18.070*
sqrt(per))/4 else 0; #a dummy function indicating impact of not owning a clothes washer
param  senior;  # a dummy parameter for senior = 1 if this is a senior-only household, = 0
otherwise
param  nopay;   # a dummy parameter = 1 if household does not pay for hot water, = 0
otherwise
param  a0{SCHEDULES};
param  a1{SCHEDULES};
param  a2{SCHEDULES};
param  a3{SCHEDULES};
param  a4{SCHEDULES};
param  a5{SCHEDULES};
param  a6{SCHEDULES};
param  a7{SCHEDULES};
param  a8{SCHEDULES};
param  a9{SCHEDULES};
param  a10{SCHEDULES};
param  a11{SCHEDULES};
param  a12{SCHEDULES};
param  a13{SCHEDULES};
param  HWL{j in SCHEDULES}:=
(a0[j]+a1[j]*per+a2[j]*age1+a3[j]*age2+a4[j]*age3+a5[j]*therm+a6[j]*tankz+a7[j]*wtmp+a8[j]*at
mp+a9[j]*athome+a10[j]*spring
                             +a11[j]*summer+a12[j]*fall+a13[j]*winter-noDWeq[j]*noDW-
noCWeq[j]*noCW)*(1-0.621*senior)*(1+0.3625*nopay); # Hot Water usage Level at each hr of time
schedule
```

```
#############
# Variables #
#############

var DECISIONS {i in DEVICES,j in SCHEDULES} binary   ,  <= if ((i = 'AC')             and ( j
< ACEarlTime          or j > ACLateTime))
                                                          or ((i = 'Heating')         and ( j

< ACEarlTime          or j > ACLateTime))
                                                          or ((i = 'Waterheater')     and ( j

< WaterheaterEarlTime or j > WaterheaterLateTime))
                                                          or ((i = 'Fridge')          and ( j

< FridgeEarlTime      or j > FridgeLateTime))
                                                          or ((i = 'DR' )             and ( j

< PVEarlTime          or j > PVLateTime))
                                                          or ((i = 'CR')              and ( j

< PVEarlTime          or j > PVLateTime))
                                                          or ((i = 'Poolpump')        and ( j

< PoolpumpEarlTime    or j > PoolpumpLateTime))
                                                          or ((i = 'Stove')           and ( j

< StoveEarlTime       or j > StoveLateTime))
                                                          or ((i = 'Lighting1')       and ( j

< LightEarlTime       or j > LightLateTime))
                                                          or ((i = 'Lighting2')       and ( j

< LightEarlTime       or j > LightLateTime))
                                                          or ((i = 'Lighting3')       and ( j

< LightEarlTime       or j > LightLateTime))
                                                          or ((i = 'Lighting4')       and ( j

< LightEarlTime       or j > LightLateTime))
                                                          or ((i = 'Lighting5')       and ( j

< LightEarlTime       or j > LightLateTime))
                                                          or ((i = 'Lighting6')       and ( j

< LightEarlTime       or j > LightLateTime))
                                                          or ((i = 'Dishwasher')      and ( j

< DishwasherEarlTime  or j > DishwasherLateTime))
                                                          or ((i = 'Washer'    )      and ( j

< WasherEarlTime      or j > WasherLateTime    ))
                                                          or ((i = 'Dryer'     )      and ( j

< DryerEarlTime       or j > DryerLateTime     ))
                                                        then 0 else 1;

var TempFrig       {j in SCHEDULES} >= if j = 1 then TempFrigIC        else TempFrigDN
,<= if j = 1 then TempFrigIC        else TempFrigUP; # Fridge Temp
var TempIN         {j in SCHEDULES} >= if j = 1 then TempIC            else TempINDN
,<= if j = 1 then TempIC            else TempINUP; # AC(Indoor) Temperature
var TempWaterheater{j in SCHEDULES} >= if j = 1 then TempWaterheaterIC else TempWaterheaterDN
,<= if j = 1 then TempWaterheaterIC else TempWaterheaterUP; # Water Temperature
var StorBAT        {j in SCHEDULES} >= if j = 1 then StorBAT0          else BATMinStorage
,<= if j = 1 then StorBAT0          else BATMaxStorage; # storage level of BAT

var UStove{SCHEDULES} binary;                # Start Up of Stove at time t
var VStove{SCHEDULES} binary;                # Shutdown of Stove at time t

var UDishwasher{SCHEDULES} binary;           # Start Up of DW at time t
var VDishwasher{SCHEDULES} binary;           # SHUTDOWN Up of DW at time t
var UWasher{SCHEDULES} binary;                       # Start Up of Washer at time t
var VWasher{SCHEDULES} binary;                       # SHUTDOWN of Washer at time t
var UDryer{SCHEDULES} binary;
var VDryer{SCHEDULES} binary;


#########
# Model #
#########

minimize TOTAL_PRICE: sum {j in SCHEDULES} (       DECISIONS['AC'         ,j] *POWER['AC']
*PRICE[j]
```

```
                                                 + DECISIONS['Heating'    ,j]
*POWER['Heating']       *PRICE[j]
                                                 + DECISIONS['Fridge'     ,j]
*POWER['Fridge']        *PRICE[j]
                                                 + DECISIONS['Waterheater',j]
*POWER['Waterheater']   *PRICE[j]
                                                 + DECISIONS['Poolpump'   ,j]
*POWER['Poolpump']      *PRICE[j]
                                                 + DECISIONS['Stove'      ,j] *POWER['Stove']
*PRICE[j]
                                                 + DECISIONS['Dryer'      ,j] *POWER['Dryer']
*PRICE[j]
                                                 + DECISIONS['Washer'     ,j]
*POWER['Washer']        *PRICE[j]
                                                 + DECISIONS['Dishwasher' ,j]
*POWER['Dishwasher']    *PRICE[j]
                                                 - DECISIONS['DR'         ,j] *POWER['DR']
*PRICE[j]
                                                 - (PVgen[j]-DECISIONS['CR',j]*BATCharge[j])
*PRICE[j]
                                           );

/*

minimize TOTAL_PRICE:
        sum {j in SCHEDULES}
                ( sum{i in DEVICES : i != 'ESD' and i !='CHAR' and i!='Lighting1' and
i!='Lighting2' } DECISIONS[i,j]*POWER[i]*PRICE[j]
                      - sum{i in DEVICES : i = 'ESD'} DECISIONS[i,j]*POWER[i]*PRICE[j]
                        - sum{i in DEVICES : i = 'CHAR'} (ESDChargeRate[j]-
DECISIONS[i,j]*ESDCharge[j])*PRICE[j]

                )
;


*/

############# AC/Heating #######################


        Const_AC1 {j in SCHEDULES, i in DEVICES: (i = 'AC' or i='Heating') and (j !=1 ) }:

        - TempIN[j]
        + TempIN[j-1] + BetaAC * ActLevA[j] - DECISIONS['AC',j] * AlfaAC +
DECISIONS['Heating', j] * AlfaHeating
         + (OutTemp[j-1] - TempIN[j-1] )*LambAC = 0;

        Const_AC2 {j in SCHEDULES, i in DEVICES: (i = 'AC' or i='Heating') }:
                DECISIONS['AC',j] + DECISIONS['Heating',j] <= 1;


         Const_AC3 {j in SCHEDULES, i in DEVICES: (i = 'AC') }:
                DECISIONS['AC',j] = 0 ;      # in winter cooling should be OFF
/*

         Const_AC3 {j in SCHEDULES, i in DEVICES: (i = 'Heating') }:
                DECISIONS['Heating',j] = 0;  # in summer heating should be OFF

*/

############# Waterheater ################


        Const_Waterheater1 {j in SCHEDULES, i in DEVICES: i = 'Waterheater' and j!=1 }:

        - TempWaterheater[j]
```

```
        + TempWaterheater[j-1] - BetaWaterheater * HWL[j] + DECISIONS[i,j] * AlfaWaterheater -
GamaWaterheater = 0;


############# Fridge ######################


        Const_Fridge1 {j in SCHEDULES, i in DEVICES: i = 'Fridge' and j!=1 }:             #
TempFrig lower limit
        - TempFrig[j] + TempFrig[j-1] + BetaFrig * ActLevB[j] - DECISIONS[i,j] * AlfaFrig +
GamaFrig = 0;


############# Stove #######################

        Const_Stove1 {i in DEVICES: i = 'Stove'} :                                        #
operation hrs of Stove is 6 hrs
                sum {j in SCHEDULES} DECISIONS[i,j] = StovROT;

        Const_Stove2 {i in DEVICES,j in SCHEDULES : i = 'Stove' and  (j >= StoveEarlTime-1 )
and (j <= StoveLateTime+1) } :
                UStove[j] - VStove[j] = DECISIONS[i,j] -DECISIONS[i,j-1] ;

         Const_Stove3 {i in DEVICES,j in SCHEDULES : i = 'Stove' and  (j >= StoveEarlTime-1 )
and (j <= StoveLateTime+1) } :
                UStove[j] + VStove[j] <= 1 ;

        Const_Stove4 {i in DEVICES,j in SCHEDULES : i = 'Stove' and (j >= StoveEarlTime-
1+StoveMinUP) and (j <= StoveLateTime+1) } :
                sum {k in j-StoveMinUP+1..j-1} UStove[k] <= DECISIONS[i,j]; # (Turn on
inequality) Min. Up time is StoveMinUP (1 hr)

        Const_Stove5 {i in DEVICES,j in SCHEDULES : i = 'Stove' and (j >= StoveEarlTime and j
<= StoveLateTime - StovMSOT)} :
                sum {k in j..j+StovMSOT} DECISIONS[i,k] <= StovMSOT ;       # Max. Up time is
4 hrs

############# Poolpump #######################


        Const_Poolpump1 {i in DEVICES: i = 'Poolpump'} :
        # operation hrs of Poolpump is "PoolpumpROT" hrs
                sum {j in SCHEDULES} DECISIONS[i,j] = PoolpumpROT;

########### PV BATT Storage Device ###########


        Const_ESD1 {j in SCHEDULES, i in DEVICES: i = 'DR' and j!=1 }:
                - StorBAT[j] + StorBAT[j-1] - DECISIONS[i,j]* POWER[i] + DECISIONS['CR',j]*
BATCharge[j] = 0;


############# Dishwasher #################


         Const_Dishwasher1 {i in DEVICES,j in SCHEDULES : i = 'Dishwasher' and (j >=
DishwasherEarlTime-1)and (j <= DishwasherLateTime+1)  } :
                UDishwasher[j]- VDishwasher[j] = DECISIONS[i,j] - DECISIONS[i,j-1] ;

         Const_Dishwasher2 {i in DEVICES,j in SCHEDULES : i = 'Dishwasher' and (j >=
DishwasherEarlTime-1 )and (j <= DishwasherLateTime+1)  } :
                UDishwasher[j] + VDishwasher[j] <= 1 ;

        Const_Dishwasher3 {i in DEVICES,j in SCHEDULES : i = 'Dishwasher' and (j >=
DishwasherEarlTime-1 and j <= DishwasherLateTime - DWMSOT)} :
                sum {k in j..j+DWMSOT} DECISIONS[i,k] <= DWMSOT ;                      # Max.
successive Up time is DWMSOT
```

```
        Const_Dishwasher4 {i in DEVICES,j in SCHEDULES : i = 'Dishwasher' and (j
>=DishwasherEarlTime + DishwasherMinUP-1) and (j <= DishwasherLateTime+1)} :
               sum {k in j-DishwasherMinUP+1..j-1} UDishwasher[k] <= DECISIONS[i,j];
# Min. Up time is DishwasherMinUP

       Const_Dishwasher5 {i in DEVICES,j in SCHEDULES : i = 'Dishwasher' and (j
>=DishwasherEarlTime + DishwasherMinDN ) and (j <= DishwasherLateTime+1)} :
               sum {k in j-DishwasherMinDN+1..j-1} VDishwasher[k] <= 1 - DECISIONS[i,j];  #
Min. Down time is DishwasherMinDN

       Const_Dishwaher6 { i in DEVICES: i = 'Dishwasher'}:
               sum {j in SCHEDULES: j >= DishwasherEarlTime and j <= DishwasherLateTime}
DECISIONS['Dishwasher',j] = DWMSOT; # total operating DWMSOT per day is 2hurs

############# B-Washer ######################


        Const_Washer1 {i in DEVICES,j in SCHEDULES : i = 'Washer' and (j >= WasherEarlTime-1
) and (j <= WasherLateTime+1) } :
               UWasher[j] - VWasher[j] = DECISIONS[i,j] - DECISIONS[i,j-1] ;

        Const_Washer2 {i in DEVICES,j in SCHEDULES : i = 'Washer' and (j >= WasherEarlTime-1
) and (j <= WasherLateTime+1) } :
               UWasher[j] + VWasher[j] <= 1 ;

       Const_Washer3 {i in DEVICES,j in SCHEDULES : i = 'Washer' and (j >= WasherEarlTime and
j <= WasherLateTime - WMSOT)} :
               sum {k in j..j+WMSOT} DECISIONS[i,k] <= WMSOT ;
# Max. Up time is WMSOT hrs

       Const_Washer4 {i in DEVICES,j in SCHEDULES : i = 'Washer' and (j >=WasherEarlTime +
WasherMinUP-1) and (j <= WasherLateTime+1)} :
               sum {k in j-WasherMinUP+1..j-1} UWasher[k] <= DECISIONS[i,j] ;
# Min. Up time is WasherMinUP

       Const_Washer5 {i in DEVICES,j in SCHEDULES : i = 'Washer' and (j >=WasherEarlTime +
WasherMinDN-1) and (j <= WasherLateTime+1)} :
               sum {k in j-WasherMinDN+1..j-1} VWasher[k] <= 1 - DECISIONS[i,j] ;         #
Min. Down time is WasherMinDN

       Const_Washer6 { i in DEVICES: i = 'Washer'}:
               sum {j in SCHEDULES: j >= WasherEarlTime and j <= WasherLateTime}
DECISIONS['Washer',j] = WMSOT; # total operating WMSOT hours per day is WasherMinUP


############# B-Dryer ######################


        Const_Dryer1 {i in DEVICES,j in SCHEDULES : i = 'Dryer' and (j >= WasherEarlTime-1 )
and (j <= DryerLateTime+1) } :
               UDryer[j]- VDryer[j] = DECISIONS[i,j] -DECISIONS[i,j-1] ;

       Const_Dryer2 {i in DEVICES,j in SCHEDULES : i = 'Dryer' and (j >= WasherEarlTime-1 )
and (j <= DryerLateTime+1)} :
               UDryer[j] + VDryer[j] <= 1 ;

       Const_Dryer3 {i in DEVICES,j in SCHEDULES : i = 'Dryer' and j >= (WasherEarlTime +
WasherMinUP-1) and j <= (DryerLateTime+1) } :
               sum {k in j-WasherMinUP+1..j-1} UDryer[k] <= DECISIONS[i,j];         #Min. Up
time is WasherMinUP

       Const_Dryer4 {i in DEVICES,j in SCHEDULES : i = 'Dryer' and j >= (WasherEarlTime +
WasherMinDN-1 ) and j <= (DryerLateTime+1) } :
               sum {k in j-WasherMinDN+1..j-1} VDryer[k] <= 1 - DECISIONS[i,j] ; #Min. Down
time is WasherMinDN
```

```
######## B- Washer/Dryer Coordination #########

# These constraints garantee that Dryer operates after Washer, and in a time less than
MTGapWasherDryer of operation of Washer

        Const_WaherDryer1 {j in SCHEDULES, i in DEVICES: (j >= DryerEarlTime and j <=
DryerLateTime) and (i = 'Dryer' or i='Washer') }:
                DECISIONS['Dryer',j] <=  sum {k in 1..MTGapWashDryer} DECISIONS['Washer',j-k];
# max. TIME GAP between dryer and washer is MTGapWashDryer

        Const_WaherDryer2 {j in SCHEDULES, i in DEVICES: (i = 'Dryer' or i='Washer') and (j >=
DryerEarlTime-1) and (j <= DryerLateTime+1)}:
                DECISIONS['Dryer',j] + DECISIONS['Washer',j] <= 1; # dryer and washer should
not operate at the same time


        Const_WaherDryer3 { i in DEVICES: i = 'Dryer' or i='Washer'}:       #  If Washer
Operates, Dryer Must operate in the same day.
                sum {j in SCHEDULES: j >= DryerEarlTime and j <= DryerLateTime}
DECISIONS['Dryer',j] = sum {j in SCHEDULES: j >= WasherEarlTime and j <= WasherLateTime}
DECISIONS['Washer',j];



#########

solve;

#########
# Print #
#########

printf " \n"   ;


printf "TOTAL_PRICE($) = %6.2f " ,
         sum {j in SCHEDULES} (       DECISIONS['AC'         ,j]  *POWER['AC']
*PRICE[j]
                                    + DECISIONS['Heating'    ,j]  *POWER['Heating']
*PRICE[j]
                                    + DECISIONS['Fridge'     ,j]  *POWER['Fridge']
*PRICE[j]
                                    + DECISIONS['Waterheater',j]  *POWER['Waterheater']
*PRICE[j]
                                    + DECISIONS['Poolpump'   ,j]  *POWER['Poolpump']
*PRICE[j]
                                    + DECISIONS['Stove'      ,j]  *POWER['Stove']
*PRICE[j]
                                    + DECISIONS['Dryer'      ,j]  *POWER['Dryer']
*PRICE[j]
                                    + DECISIONS['Washer'     ,j]  *POWER['Washer']
*PRICE[j]
                                    + DECISIONS['Dishwasher' ,j]  *POWER['Dishwasher']
*PRICE[j]
                                    + Illum1[                j]  *POWER['Lighting1']
*PRICE[j]
                                    + Illum2[                j]  *POWER['Lighting2']
*PRICE[j]
                                    + Illum3[                j]  *POWER['Lighting3']
*PRICE[j]
                                    + Illum4[                j]  *POWER['Lighting4']
*PRICE[j]
                                    + Illum5[                j]  *POWER['Lighting5']
*PRICE[j]
```

```
                                    + Illum6[                 j]  *POWER['Lighting6']
*PRICE[j]
                                    - DECISIONS['DR'         ,j]  *POWER['DR']
*PRICE[j]
                                    - (PVgen[j]-DECISIONS['CR',j] * BATCharge[j])
*PRICE[j]

                                ) / 400000 ;


printf " \n"   ;



printf "TOTAL_consumption(KWH) = %6.2f " ,
          sum {j in SCHEDULES} (       DECISIONS['AC'         ,j]  *POWER['AC']
                                     + DECISIONS['Heating'    ,j]  *POWER['Heating']
                                     + DECISIONS['Fridge'     ,j]  *POWER['Fridge']
                                     + DECISIONS['Waterheater',j]  *POWER['Waterheater']
                                     + DECISIONS['Poolpump'   ,j]  *POWER['Poolpump']
                                     + DECISIONS['Stove'      ,j]  *POWER['Stove']
                                     + DECISIONS['Dryer'      ,j]  *POWER['Dryer']
                                     + DECISIONS['Washer'     ,j]  *POWER['Washer']
                                     + DECISIONS['Dishwasher' ,j]  *POWER['Dishwasher']
                                     + Illum1[                 j]  *POWER['Lighting1']
                                     + Illum2[                 j]  *POWER['Lighting2']
                                     + Illum3[                 j]  *POWER['Lighting3']
                                     + Illum4[                 j]  *POWER['Lighting4']
                                     + Illum5[                 j]  *POWER['Lighting5']
                                     + Illum6[                 j]  *POWER['Lighting6']
                                     - DECISIONS['DR'         ,j]  *POWER['DR']
                                     - (PVgen[j]-DECISIONS['CR',j] * BATCharge[j])
                                 ) / 4000 ;


printf "\n"    ;

printf "\n"    ;
printf " Appliances Daily Consumption (Kwh)\n"      ;
printf "-----------------------------------------------------------------------------------
------------ \n"      ;
printf "   AC     WH     Frg   Stov   Heat  P.pump   DR    CR   Lit1   Lit2   DW     Wash
Dryer |\n"     ;
printf "-----------------------------------------------------------------------------------
------------|\n"      ;


       for {i in DEVICES}  printf "%7.2f" , sum {j in SCHEDULES} (DECISIONS[i,j]
*POWER[i]/4000) ;


printf "\n"    ;
printf "\n"    ;

printf " Appliances Daily Costs (cents)\n"   ;
printf "-----------------------------------------------------------------------------------
- \n"   ;
printf "   AC    WH    Frg  Stov  Heat P.pump   DR    CR Lit1  Lit2   DW Wash    Dryer |\n"
       ;
printf "-----------------------------------------------------------------------------------
-|\n"   ;


       for {i in DEVICES}  printf "%6.1f" , sum {j in SCHEDULES} (DECISIONS[i,j] *POWER[i]*
PRICE[j]/4000) ;
```

```
printf "\n"    ;
printf "\n"    ;


printf "Lighting_consumption(KWH) = %6.2f ", (sum {j in SCHEDULES , i in DEVICES : i =
'Lighting1'} ( Illum1[j]*POWER[i]) / 4000)
                                              + (sum {j in SCHEDULES , i in DEVICES : i =
'Lighting2'} ( Illum2[j]*POWER[i]) / 4000)
                                              + (sum {j in SCHEDULES , i in DEVICES : i =
'Lighting3'} ( Illum3[j]*POWER[i]) / 4000)
                                              + (sum {j in SCHEDULES , i in DEVICES : i =
'Lighting4'} ( Illum4[j]*POWER[i]) / 4000)
                                              + (sum {j in SCHEDULES , i in DEVICES : i =
'Lighting5'} ( Illum5[j]*POWER[i]) / 4000)
                                              + (sum {j in SCHEDULES , i in DEVICES : i =
'Lighting6'} ( Illum6[j]*POWER[i]) / 4000);
printf "\n" ;
printf "hotwater_consumption(liters) = %6.2f ", sum {j in SCHEDULES } ( HWL[j]/4) ;
printf "\n"    ;
printf "ESD_injection(KWH) = %6.2f ", sum {j in SCHEDULES , i in DEVICES : i = 'DR'} (
DECISIONS[i,j]*POWER[i]) / 4000 ;
printf "\n" ;
printf "PV_direct_injection(KWH) = %6.2f ", sum {j in SCHEDULES , i in DEVICES : i = 'CR'}
(PVgen[j]-DECISIONS['CR',j]*BATCharge[j]) / 4000 ;
printf "\n" ;
printf "PV_total_injection(KWH) = %6.2f ", sum {j in SCHEDULES , i in DEVICES : i = 'CR'}
(PVgen[j]-DECISIONS['CR',j]*BATCharge[j]) / 4000
                                            + sum {j in SCHEDULES , i in DEVICES : i = 'DR'}
( DECISIONS[i,j]*POWER[i]) / 4000 ;
printf "\n"    ;
printf "\n"    ;

printf "Lighting_price = %6.2f ", (sum {j in SCHEDULES , i in DEVICES : i = 'Lighting1'} (
Illum1[j]*POWER[i]*PRICE[j]) / 4000)
                                + (sum {j in SCHEDULES , i in DEVICES : i = 'Lighting2'} (
Illum2[j]*POWER[i]*PRICE[j]) / 4000)
                                + (sum {j in SCHEDULES , i in DEVICES : i = 'Lighting3'} (
Illum3[j]*POWER[i]*PRICE[j]) / 4000)
                                + (sum {j in SCHEDULES , i in DEVICES : i = 'Lighting4'} (
Illum4[j]*POWER[i]*PRICE[j]) / 4000)
                                + (sum {j in SCHEDULES , i in DEVICES : i = 'Lighting5'} (
Illum5[j]*POWER[i]*PRICE[j]) / 4000)
                                + (sum {j in SCHEDULES , i in DEVICES : i = 'Lighting6'} (
Illum6[j]*POWER[i]*PRICE[j]) / 4000);
printf "\n"    ;
printf "BAT_saving = %6.2f ", sum {j in SCHEDULES , i in DEVICES : i = 'DR'} (
DECISIONS[i,j]*POWER[i]*PRICE[j]) / 4000 ;
printf "\n" ;
printf "PV_direct_saving(KWH) = %6.2f ", sum {j in SCHEDULES , i in DEVICES : i = 'CR'}
((PVgen[j]-DECISIONS['CR',j]*BATCharge[j])*PRICE[j]) / 4000 ;
printf "\n" ;
printf "PV_TOTAL_saving(KWH) = %6.2f ", sum {j in SCHEDULES , i in  DEVICES : i = 'CR'}
((PVgen[j]-DECISIONS['CR',j]*BATCharge[j])*PRICE[j]) / 4000
                                        +sum {j in SCHEDULES , i in DEVICES : i = 'DR'} (
DECISIONS[i,j]*POWER[i]*PRICE[j]) / 4000 ;
printf "\n" ;



printf "-------------------------------------------------------------------------------------
------------\n"       ;
printf "| Time|   AC  WH Frg Stov Heat P.pump  DR   CR  Lit1 Lit2 Lit3 Lit4 Lit5 Lit6  DW
Wash Dryer|\n" ;
printf "-------------------------------------------------------------------------------------
------------|\n"      ;
```

```
for{j in SCHEDULES} {
                       printf "%1s %3d %1s","|", j, "|" ;
                       for {i in DEVICES}  printf "%5d" , DECISIONS[i,j] ;
                        printf "%5s \n", "    |"   ;

                   }

printf "--------------------------------------------------------------------------------\n"
        ;


printf "-------------------------------------------------------------------\n"     ;

printf "Time   T_out      T_WH    T_Fr    Tempin    HWL  noDW  noCW   Act-Lev PRICE\n";

for{j in SCHEDULES} {
               printf "%3d %8.2f %8.2f %8.2f %8.2f %8.2f %8.2f %8.2f %8.2f %8.2f\n", j
,OutTemp[j], TempWaterheater[j], TempFrig[j],TempIN[j],HWL[j],noDWeq[j],noCWeq[j],ActLevA[j],
PRICE[j]
                       ;
                   }

printf "-------------------------------------------------------------------\n"     ;




printf "\n"    ;


printf "-------------------------------------------------------------------------------------
--------\n" ;

printf "Time  OUTiLLum1  ActLevB       EEC       EEC1        ActLevA  \n";

for{j in SCHEDULES} {
               printf " %3d %8.2f %10.2f %10.2f %10.2f %10.2f
\n",j,OutIllum1[j],ActLevB[j],EEC[j],EEC1[j], ActLevA[j]
                       ;
                   }

printf "-------------------------------------------------------------------------------------
----------\n";

printf "-------------------------------------------------------------------------------------
--------\n" ;

printf "Time  illum1  illum2  illum3 illum4  illum5  illum6   \n";

for{j in SCHEDULES} {
               printf " %3d %10.2f %10.2f %10.2f %10.2f %10.2f %10.2f
\n",j,Illum1[j],Illum2[j],Illum3[j],Illum4[j],Illum5[j],Illum6[j]
                       ;
                   }

printf "-------------------------------------------------------------------------------------
----------\n";


printf "-------------------------------------------------------------------\n"     ;

printf "Time   BATstorage   DR     CR   PV-Gen   chargingPower  PRICE\n";

for{j in SCHEDULES} {
               printf "%3d %12.2f %6.2f %6.2f %8.2f %8.2f %8.2f\n", j ,
StorBAT[j],DECISIONS['DR',j],DECISIONS['CR',j],PVgen[j],BATCharge[j],PRICE[j]
```

```
                       ;
                   }

printf "-------------------------------------------------------------------\n"     ;

printf "-----------------------\n"    ;

printf "Time POWERCONSUM\n";

for{j in SCHEDULES} {
        printf "%3d %8.2f\n", j,      (DECISIONS['AC'        ,j]  *POWER['AC']
                                     + DECISIONS['Heating'    ,j]  *POWER['Heating']
                                     + DECISIONS['Fridge'     ,j]  *POWER['Fridge']
                                     + DECISIONS['Waterheater',j]  *POWER['Waterheater']
                                     + DECISIONS['Poolpump'   ,j]  *POWER['Poolpump']
                                     + DECISIONS['Stove'      ,j]  *POWER['Stove']
                                     + DECISIONS['Dryer'      ,j]  *POWER['Dryer']
                                     + DECISIONS['Washer'     ,j]  *POWER['Washer']
                                     + DECISIONS['Dishwasher' ,j]  *POWER['Dishwasher']
                                     + Illum1[                 j]  *POWER['Lighting1']
                                     + Illum2[                 j]  *POWER['Lighting2']
                                     + Illum3[                 j]  *POWER['Lighting3']
                                     + Illum4[                 j]  *POWER['Lighting4']
                                     + Illum5[                 j]  *POWER['Lighting5']
                                     + Illum6[                 j]  *POWER['Lighting6']
                                     - DECISIONS['DR'         ,j]  *POWER['DR']
                                     - (PVgen[j]-DECISIONS['CR',j] * BATCharge[j])
                                 ) / 4000
                       ;
                   }

printf "----------------------\n"     ;
```

```
#Data file:

set DEVICES := AC Waterheater Fridge Stove Heating Poolpump DR CR Lighting1 Lighting2
Lighting3 Lighting4 Lighting5 Lighting6 Dishwasher Washer Dryer;
param POWER :=
       AC             11500
       Waterheater    3600
       Fridge         250
       Stove          1500
       Heating        400
       Poolpump       750
       DR             100  # discharge power rating of ESD device ( DC/AC inverter) in watts
        CR              55   # battery charger power rating of PV system
       Lighting1      0.3  # we are assuming that each 30 watts generate 100 Lux in zone1
        Lighting2       1    # we are assuming that each 100 watts generate 100 Lux in zone2
       Lighting3       0.6  # we are assuming that each 60 watts generate 100 Lux in zone1
        Lighting4       1    # we are assuming that each 100 watts generate 100 Lux in zone2
       Lighting5      0.6  # we are assuming that each 60 watts generate 100 Lux in zone1
        Lighting6       1    # we are assuming that each 100 watts generate 100 Lux in zone2
       Dishwasher     600  # dishwasher average consumption is 1200 watts per cycle of 2
hours
       Washer         450  # washer average consumption is 900 watts per cycle of 2 hours
       Dryer          1100 # dryer average consumption is 2200 watts per cycle of 2 hours
;



param DishwasherEarlTime    :=       64     ;
param DishwasherLateTime    :=       92     ;
param DishwasherMinUP       :=  8      ;
param DishwasherMinDN       :=  4      ;
param DWMSOT                :=  8      ;

param WasherEarlTime        :=  64     ;
param WasherLateTime        :=  92     ;
param  WasherMinUP         :=  8      ;
param  WasherMinDN         :=  4      ;
param   WMSOT               :=  8      ;

param DryerEarlTime         :=       64     ;
param DryerLateTime         :=       92     ;
param  DryerMinUP          :=  8      ;
param  DryerMinDN          :=  4      ;

param MTGapWashDryer        :=  12     ;

param TempFrigUP            :=  8      ;
param TempFrigDN            :=  2      ;
param TempFrigIC            :=  5      ;
param BetaFrig              :=  0.28   ;
param AlfaFrig              :=  5.5    ;
param GamaFrig              :=  1.21   ;
param FridgeEarlTime        :=  1      ;
param FridgeLateTime        :=  96     ;

param TempWaterheaterUP     :=  65     ;
param TempWaterheaterDN     :=  55     ;
param TempWaterheaterIC     :=  60     ;
param BetaWaterheater       :=  0.068  ;
param AlfaWaterheater       :=  4.44   ;
param GamaWaterheater       :=  0.083  ;
param WaterheaterEarlTime   :=       1      ;
param WaterheaterLateTime   :=       96     ;

param StoveMinUP            :=  4      ;
param StovMSOT              :=  16     ;
param StovROT               :=  24     ;
```

```
param TempINUP              :=  23     ;
param TempINDN              :=  17     ;
param TempIC                :=  20     ;
param AlfaHeating           :=  1.02   ;
param AlfaAC                :=  0.51   ;
param BetaAC                :=  0.0275 ;
param LambAC                :=  0.0075 ;
param ACEarlTime            :=  1      ;
param ACLateTime            :=  96     ;

param PoolpumpROT           :=  32     ;
param PoolpumpEarlTime      :=  1      ;
param PoolpumpLateTime      :=  96     ;

param PVEarlTime            :=  2      ;
param PVLateTime            :=  96     ;
param BATMinStorage         :=  250    ;
param BATMaxStorage         :=  2000   ;
param StorBAT0              :=  250    ;


param   StoveEarlTime      :=  20     ;
param   StoveLateTime      :=        96     ;
param   LightEarlTime      :=  1      ;
param   LightLateTime      :=  96     ;

param  per      := 4     ;    # number of persons in a house
param  age1     := 1     ;    # number of preschool children (0-5 yrs)
param  age2     := 1     ;    # number of school age children (6-13 yrs)
param  age3     := 2     ;    # number of adults (14 yrs and over)
param  therm    := 60    ;    # water heater thermostat setting, (°C)
param  tankz    := 185   ;    # water heater nominal tank size, (Liters)
param  wtmp     := 8     ;    # water heater inlet water temperature, (°C)
param  atmp     := -10   ;    # outdoor air ambient temperature,average on each day (°C)
param  athome   := 1     ;    # a dummy parameter for the presence of adults at home during
day (unemployed person)
param  spring   := 0     ;    # a dummy parameter for Spring (1 if “spring”, zero otherwise)
param  summer   := 0     ;    # a dummy parameter for Summer (1 if “spring”, zero otherwise)
param  fall     := 0     ;    # a dummy parameter for fall (1 if “spring”, zero otherwise)
param  winter   := 1     ;    # a dummy parameter for winter (1 if “spring”, zero otherwise)
param  noDW     := 0     ;    # a dummy parameter for dishwasher (1 if dishwasher is not
available, zero otherwise)
param  noCW     := 0     ;    # a dummy parameter for clothes washer (1 if washer is not
available, zero otherwise)
param  senior   := 0     ;    # a dummy parameter for senior = 1 if this is a senior-only
household, = 0 otherwise
param  nopay    := 0     ;    # a dummy parameter = 1 if household does not pay for hot
water, = 0 otherwise


param:

SCHEDULES:     EEC    PVgen   OutTemp ReqIllum6     OutIllum6      ReqIllum5      OutIllum5
        ReqIllum4     OutIllum4      ReqIllum3      OutIllum3       ReqIllum2     OutIllum2
       ReqIllum1      OutIllum1      PRICE  a0     a1      a2     a3      a4     a5      a6
       a7      a8     a9      a10    a11     a12    a13:=

1      150     0      -17     0      0       0      0       0      0       0      0       0
       0       0      0       4.4    0      0.6163 0      0       0      0       -0.0017 0
       0       0      0       0      0       0.5523
2      150     0      -17     0      0       0      0       0      0       0      0       0
       0       0      0       4.4    0      0.6163 0      0       0      0       -0.0017 0
       0       0      0       0      0       0.5523
3      150     0      -17     0      0       0      0       0      0       0      0       0
       0       0      0       4.4    0      0.6163 0      0       0      0       -0.0017 0
       0       0      0       0      0       0.5523
```

```
4       150     0       -17     0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
5       100     0       -14.3   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
6       100     0       -14.3   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
7       100     0       -14.3   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
8       100     0       -14.3   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
9       87.5    0       -15.6   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
10      87.5    0       -15.6   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
11      87.5    0       -15.6   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
12      87.5    0       -15.6   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
13      125     0       -16.5   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
14      125     0       -16.5   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
15      125     0       -16.5   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
16      125     0       -16.5   0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
17      75      0       -19     0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
18      75      0       -19     0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
19      75      0       -19     0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
20      75      0       -19     0       0       0       0       0       0       0       0       0
        0       0       0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
21      70      0       -20.1   0       1       100     3       0       1       100     3       0
        1       100     3       4.4     2.0956  0       0       3.483   7.9861  0       0.0269  -
0.5424  -0.6603 -3.6609 0       13.601  0       0
22      70      0       -20.1   0       2       100     7       0       2       100     7       0
        2       100     7       4.4     2.0956  0       0       3.483   7.9861  0       0.0269  -
0.5424  -0.6603 -3.6609 0       13.601  0       0
23      70      0       -20.1   0       4       100     13      0       4       100     13      0
        4       100     13      4.4     2.0956  0       0       3.483   7.9861  0       0.0269  -
0.5424  -0.6603 -3.6609 0       13.601  0       0
24      70      0       -20.1   0       7       100     23      0       7       100     23      0
        7       100     23      4.4     2.0956  0       0       3.483   7.9861  0       0.0269  -
0.5424  -0.6603 -3.6609 0       13.601  0       0
25      100     0       -19.6   0       10      100     33      0       10      100     33      0
        10      100     33      4.4     2.0956  0       0       3.483   7.9861  0       0.0269  -
0.5424  -0.6603 -3.6609 0       13.601  0       0
```

```
26      100     12      -19.6   0       15      100     50      0       15      100     50      0
        15      100     50      4.4     2.0956  0       0       3.483   7.9861  0       0.0269  -
0.5424  -0.6603 -3.6609 0       13.601  0       0
27      100     12      -19.6   0       20      100     67      0       20      100     67      0
        20      100     67      4.4     2.0956  0       0       3.483   7.9861  0       0.0269  -
0.5424  -0.6603 -3.6609 0       13.601  0       0
28      100     24      -19.6   0       25      100     83      0       25      100     83      0
        25      100     83      4.4     2.0956  0       0       3.483   7.9861  0       0.0269  -
0.5424  -0.6603 -3.6609 0       13.601  0       0
29      262.5   12      -23.3   0       30      100     100     0       30      100     100     0
        30      100     100     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
30      262.5   12      -23.3   0       40      100     133     0       40      100     133     0
        40      100     133     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
31      262.5   24      -23.3   0       50      100     133     0       50      100     167     0
        50      100     167     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
32      262.5   48      -23.3   0       60      100     133     0       60      100     150     0
        60      100     200     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
33      150     72      -21.5   100     65      100     133     100     65      100     125     100
        65      100     217     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
34      150     143     -21.5   100     70      100     133     100     70      100     100     100
        70      100     233     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
35      150     167     -21.5   100     75      100     133     100     75      100     80      100
        75      100     250     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
36      150     191     -21.5   100     80      100     133     100     80      100     70      100
        80      100     267     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
37      200     48      -21     100     85      100     120     100     85      100     60      100
        85      100     283     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
38      200     24      -21     100     90      100     120     100     90      100     50      100
        90      100     300     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
39      200     24      -21     100     92      100     118     100     92      100     30      100
        92      100     307     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
40      200     24      -21     100     94      100     118     100     94      100     20      100
        94      100     313     9.3     0       0       1.0853  1.5331  2.4972  0       0       0
        0       9.0418  0       0       -1.6353 2.1403
41      350     24      -16.9   100     95      100     110     100     95      100     20      100
        95      100     317     9.3     -0.3876 0       0.9668  1.0849  2.0956  -0.0218 0       0
        0       6.1986  0       0       -1.6834 1.5187
42      350     96      -16.9   100     96      100     110     100     96      100     20      100
        96      100     320     9.3     -0.3876 0       0.9668  1.0849  2.0956  -0.0218 0       0
        0       6.1986  0       0       -1.6834 1.5187
43      350     191     -16.9   100     97      100     110     100     97      100     20      100
        97      100     323     9.3     -0.3876 0       0.9668  1.0849  2.0956  -0.0218 0       0
        0       6.1986  0       0       -1.6834 1.5187
44      350     191     -16.9   100     98      100     100     100     98      100     20      100
        98      100     327     9.3     -0.3876 0       0.9668  1.0849  2.0956  -0.0218 0       0
        0       6.1986  0       0       -1.6834 1.5187
45      800     191     -14     100     100     100     100     100     100     100     20      100
        100     100     330     8       -0.3876 0       0.9668  1.0849  2.0956  -0.0218 0       0
        0       6.1986  0       0       -1.6834 1.5187
46      800     191     -14     100     110     100     100     100     110     100     20      100
        110     100     333     8       -0.3876 0       0.9668  1.0849  2.0956  -0.0218 0       0
        0       6.1986  0       0       -1.6834 1.5187
47      800     215     -14     100     110     100     90      100     110     100     0       100
        110     100     333     8       -0.3876 0       0.9668  1.0849  2.0956  -0.0218 0       0
        0       6.1986  0       0       -1.6834 1.5187
```

```
48      800     191     -14     100     110     100     90      100     110     100     0       100
        110     100     333     8       -0.3876 0       0.9668  1.0849  2.0956  -0.0218 0       0
        0       6.1986  0       0       -1.6834 1.5187
49      1100    167     -13     100     110     100     90      100     110     100     0       100
        110     100     333     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
50      1100    119     -13     100     110     100     80      100     110     100     0       100
        110     100     333     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
51      1100    0       -13     100     110     100     70      100     110     100     0       100
        110     100     333     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
52      1100    119     -13     100     110     100     70      100     110     100     0       100
        110     100     333     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
53      150     239     -12.8   100     110     100     50      100     110     100     0       100
        110     100     333     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
54      150     239     -12.8   100     110     100     50      100     110     100     0       100
        110     100     333     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
55      150     239     -12.8   100     110     100     40      100     110     100     0       100
        110     100     333     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
56      150     191     -12.8   100     110     100     30      100     110     100     0       100
        110     100     333     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
57      100     239     -13.1   100     110     100     20      100     110     100     0       100
        110     100     327     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
58      100     239     -13.1   100     125     100     10      100     125     100     0       100
        125     100     320     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
59      100     24      -13.1   100     150     100     10      100     125     100     0       100
        150     100     303     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
60      100     191     -13.1   100     75      100     10      100     125     100     0       100
        200     100     267     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
61      125     191     -13.2   100     75      100     5       100     125     100     0       100
        220     100     250     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
62      125     167     -13.2   100     75      100     5       100     125     100     0       100
        220     100     200     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
63      125     143     -13.2   100     75      100     0       100     125     100     0       100
        220     100     167     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
64      125     119     -13.2   100     75      100     0       100     125     100     0       100
        220     100     133     8       -0.2907 0       1.979   1.2     2.3072  -0.0906 0.0083  0
        -0.0743 4.0228  0       0       0       2.5854
65      225     119     -13.5   100     75      100     0       100     125     100     0       100
        210     100     100     8       0.7753  0       1.5679  2.0415  3.6018  0       0       -
0.3134  -0.357  5.3492  0       3.6855  0       3.656
66      225     96      -13.5   100     75      200     0       100     110     200     0       100
        200     200     67      8       0.7753  0       1.5679  2.0415  3.6018  0       0       -
0.3134  -0.357  5.3492  0       3.6855  0       3.656
67      225     72      -13.5   100     75      200     0       100     100     200     0       100
        100     200     50      8       0.7753  0       1.5679  2.0415  3.6018  0       0       -
0.3134  -0.357  5.3492  0       3.6855  0       3.656
68      225     48      -13.5   100     75      200     0       100     75      200     0       100
        75      200     33      8       0.7753  0       1.5679  2.0415  3.6018  0       0       -
0.3134  -0.357  5.3492  0       3.6855  0       3.656
69      187.5   48      -14.4   100     50      200     0       100     50      200     0       100
        50      200     17      9.3     0.7753  0       1.5679  2.0415  3.6018  0       0       -
0.3134  -0.357  5.3492  0       3.6855  0       3.656
```

```
70      187.5   36      -14.4   100     25      200     0       100     25      200     0       100
        25      200     10      9.3     0.7753  0       1.5679  2.0415  3.6018  0       0       -
0.3134  -0.357  5.3492  0       3.6855  0       3.656
71      187.5   36      -14.4   100     20      200     0       100     20      200     0       100
        20      200     7       9.3     0.7753  0       1.5679  2.0415  3.6018  0       0       -
0.3134  -0.357  5.3492  0       3.6855  0       3.656
72      187.5   24      -14.4   100     10      200     0       100     10      200     0       100
        10      200     3       9.3     0.7753  0       1.5679  2.0415  3.6018  0       0       -
0.3134  -0.357  5.3492  0       3.6855  0       3.656
73      562.5   24      -15.2   100     5       200     0       100     5       200     0       100
        5       200     2       9.3     4.4577  0       2.7456  4.4092  3.3455  -0.1015 0.0187  0
        -0.3523 0       0       8.0527  -3.2509 0
74      562.5   12      -15.2   100     0       200     0       100     0       200     0       100
        0       200     0       9.3     4.4577  0       2.7456  4.4092  3.3455  -0.1015 0.0187  0
        -0.3523 0       0       8.0527  -3.2509 0
75      562.5   12      -15.2   100     0       200     0       100     0       200     0       100
        0       200     0       9.3     4.4577  0       2.7456  4.4092  3.3455  -0.1015 0.0187  0
        -0.3523 0       0       8.0527  -3.2509 0
76      562.5   0       -15.2   100     0       200     0       100     0       200     0       100
        0       200     0       9.3     4.4577  0       2.7456  4.4092  3.3455  -0.1015 0.0187  0
        -0.3523 0       0       8.0527  -3.2509 0
77      187.5   0       -15.3   100     0       200     0       100     0       200     0       100
        0       200     0       9.3     4.4577  0       2.7456  4.4092  3.3455  -0.1015 0.0187  0
        -0.3523 0       0       8.0527  -3.2509 0
78      187.5   0       -15.3   100     0       200     0       100     0       200     0       100
        0       200     0       9.3     4.4577  0       2.7456  4.4092  3.3455  -0.1015 0.0187  0
        -0.3523 0       0       8.0527  -3.2509 0
79      187.5   0       -15.3   100     0       200     0       100     0       200     0       100
        0       200     0       9.3     4.4577  0       2.7456  4.4092  3.3455  -0.1015 0.0187  0
        -0.3523 0       0       8.0527  -3.2509 0
80      187.5   0       -15.3   100     0       200     0       100     0       300     0       100
        0       400     0       9.3     4.4577  0       2.7456  4.4092  3.3455  -0.1015 0.0187  0
        -0.3523 0       0       8.0527  -3.2509 0
81      212.5   0       -15.7   100     0       200     0       100     0       300     0       100
        0       400     0       9.3     4.2881  0       1.4434  3.4394  2.5135  -0.0436 0       0
        -0.2848 -2.4166 0       3.3773  -3.5511 0
82      212.5   0       -15.7   100     0       200     0       100     0       300     0       100
        0       400     0       9.3     4.2881  0       1.4434  3.4394  2.5135  -0.0436 0       0
        -0.2848 -2.4166 0       3.3773  -3.5511 0
83      212.5   0       -15.7   100     0       200     0       100     0       300     0       100
        0       400     0       9.3     4.2881  0       1.4434  3.4394  2.5135  -0.0436 0       0
        -0.2848 -2.4166 0       3.3773  -3.5511 0
84      212.5   0       -15.7   100     0       200     0       100     0       300     0       100
        0       400     0       9.3     4.2881  0       1.4434  3.4394  2.5135  -0.0436 0       0
        -0.2848 -2.4166 0       3.3773  -3.5511 0
85      250     0       -15.9   100     0       200     0       100     0       300     0       100
        0       400     0       4.4     4.2881  0       1.4434  3.4394  2.5135  -0.0436 0       0
        -0.2848 -2.4166 0       3.3773  -3.5511 0
86      250     0       -15.9   100     0       200     0       100     0       300     0       100
        0       400     0       4.4     4.2881  0       1.4434  3.4394  2.5135  -0.0436 0       0
        -0.2848 -2.4166 0       3.3773  -3.5511 0
87      250     0       -15.9   100     0       200     0       100     0       300     0       100
        0       400     0       4.4     4.2881  0       1.4434  3.4394  2.5135  -0.0436 0       0
        -0.2848 -2.4166 0       3.3773  -3.5511 0
88      250     0       -15.9   100     0       200     0       100     0       300     0       100
        0       400     0       4.4     4.2881  0       1.4434  3.4394  2.5135  -0.0436 0       0
        -0.2848 -2.4166 0       3.3773  -3.5511 0
89      75      0       -17     100     0       200     0       100     0       300     0       100
        0       400     0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
90      75      0       -17     100     0       200     0       100     0       300     0       100
        0       400     0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
91      75      0       -17     100     0       200     0       100     0       0       0       100
        0       400     0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
```

```
92      75      0       -17     100     0       200     0       100     0       0       0       100
        0       400     0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
93      75      0       -17.7   100     0       200     0       100     0       0       0       100
        0       400     0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
94      75      0       -17.7   100     0       200     0       100     0       0       0       100
        0       400     0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
95      75      0       -17.7   100     0       200     0       100     0       0       0       100
        0       400     0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
96      75      0       -17.7   100     0       200     0       100     0       0       0       100
        0       400     0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
97      75      0       -17.9   100     0       200     0       100     0       0       0       100
        0       400     0       4.4     0       0.6163  0       0       0       0       -0.0017 0
        0       0       0       0       0       0.5523
;
```

Bibliography

[1] Ontario Power Authority, "Supply Mix Advice Report", vol. 1, part 1-1, Ontario, Canada, Dec. 2005. Available online at: http://www.powerauthority.on.ca/Storage/18/1338_Part_1-1_Supply_Mix_Summary.pdf

[2] Ontario Power Authority, "Supply Mix Analysis Report", vol. 2, part 2-6, Ontario, Canada, Dec. 2005. Available online at: http://www.powerauthority.on.ca/Storage/18/1350_Part_2-6_Methodologies_and_Assumptions_Adopted.pdf

[3] Natural Resources Canada, "Comprehensive Energy Use Database Tables: Residential Sector", Ontario, Canada, 2007. Available online at: http://oee.nrcan-rncan.gc.ca/corporate/statistics/neud/dpa/trends_res_on.cfm

[4] J. R. Munoz, C. A. Canizares and K. Bhattacharya, "Current State of Conservation and Demand Management", Internal Report of Energy Hub Management Project, University of Waterloo, Ontario, Canada, 2008.

[5] US Department of Energy, "The Smart Grid: An Introduction", USA, 2010. Available online at: http://www.oe.energy.gov/DocumentsandMedia/DOE_SG_Book_Single_Pages(1).pdf

[6] Social Houses Services Corporation, "Smart Meters and Social Housing: Energy Conservation and Energy Poverty", Ontario, Canada, Aug. 2006. Available online at: http://www.shscorp.ca/shscnew/content/docs/Smart%20meters%20and%20social%20housing.pdf

[7] A. M. Asawari, "Computer Aided Home Energy Management System", MASc thesis, Electrical Engineering Department, West Virginia University, USA, 2005.

[8] University of Waterloo Energy Hub Management System Project, https://sharepoint.uwaterloo.ca/sites/Energy%20Hubs/default.aspx

[9] M. C. Bozchalui, "Optimal Operation of Energy Hubs in the Context of Smart Grids", PhD proposal, Electrical and Computer Engineering, University of Waterloo, Ontario, Canada 2009.

[10] J. T. Linderoth and T. K. Ralphs, "Non-commercial software for mixed integer linear programming", Lehigh University, 2005. Available online at: http://www.lehigh.edu/~jtl3/papers/MILP04.pdf

[11] V. Silva, V. Stanojevic, D. Pudjinato and G. Strbac, "Value of Smart Domestic Appliances in Stressed Electricity Networks: SMART-A", Imperial College, University of Manchester, London, UK, Sept. 2009. Available online at: http://www.smart-a.org/W_P_4_D_4_4_Energy_Networks_Report_final.pdf

[12] J. Hong, “The Development, Implementation, and Application of Demand Side Management and control (DSM+c) Algorithm for Integrating Microgeneration”, PhD thesis, University of Strathclyde, UK, Mar. 2009.

[13] M.M. Armstrong, M.C. Swinton, H. Ribberink, I. Beausoleil-Morrison, and J. Millette, “Synthetically derived profiles for representing occupant driven electric loads in Canadian housing”, *J. Building Performance Simulation*, vol. 2, no. 1, pp.15–30, 2009.

[14] GSW Water Heating, “Electric Water Heater Installation and Operating Instructions”, Ontario, Canada. Available online at: http://www.gsw-wh.com/pdfs/products/water/standard/61515-1K.PDF

[15] J. Lutz, X. Liu, J. McMahon, C. Dunham, L. Shown, and Q. McCure, “Modeling patterns of hot water use in households”, Lawrence Berkeley National Laboratory, 1996.Available online at: http://www.osti.gov/bridge/purl.cover.jsp;jsessionid=3B8099840DF398ACF834A7411A9F6EAA?purl=/821315-tWoV0U/native/

[16] Ladd, G.O. and J.L. Harrison, “Electric Water Heating for Single-Family Residences: Group Load Research and Analysis”, report number EA- 4006 prepared for Electric Power Research Institute. EPRI by Gilbert Associates Inc., Palo Alto, California, USA, May. 1985.

[17] B. Daryanian, “Scheduling of Electricity Consumption under Spot Prices”. PhD thesis, Massachusetts Institute of Technology, Mechanical Engineering Department, Massachusetts, USA, 1989.

[18] United Nations Environment Program, “Refrigeration and Air Conditioning System, Energy Efficiency Guide for Industry in Asia”, USA, 2006.

[19] S. Porkhial, B. Khastoo and M. Saffar-Avval, “Transient response of dry expansion evaporator in household refrigerators”, *Applied Thermal Engineering*, issue 24, pp.1465–1480, 2004.

[20] M. R. Laleman, “Sensible and Latent Energy Loading on a Refrigerator During Open Door Conditions”, MASc thesis, Department of Mechanical and Industrial Engineering, University of Illinois, 1992

[21] C. Inan, T. A. Newell, and A. N. Egrican, “Heat and Mass Transfer through a Domestic Refrigerator and Evaluation of Evaporator Performance under Frosted Conditions”, Air Conditioning and Refrigeration Center University of Illinois Mechanical & Industrial Engineering Dept., Urbana, IL, 2000. Available online at: https://www.ideals.illinois.edu/bitstream/handle/2142/13407/CR048.pdf?sequence=2

[22] F.C. McQuiston and J.D. Spitler, *Cooling and Heating Load Calculation Manual*, ASHRAE Inc., 2nd edition, 1992.

[23] R. Stamminger et al , “Synergy Potential of Smart Appliances”, Report D2.3 of WP 2 from the Smart-A project University of Bonn, March 2009. Available online at:
http://www.smart-a.org/WP2_D_2_3_Synergy_Potential_of_Smart_Appliances.pdf

[24] George Wilkenfeld and Associates Pty Ltd, “Analysis of the Potential for Energy Efficiency Measures for Domestic Swimming Pool and Spa Pool Equipment”, National Appliance and Equipment Energy Efficiency Committee (NAEEEC) and the Australian Greenhouse Office, Sept. 2004. Available online at:
http://www.energyrating.gov.au/library/pubs/200412-mepspools.pdf

[25] “Electronic energy saving lamps”, Osram dk, Available online at:
http://www.osram.dk/osram_dk/Professionelle/Brochurer_&_kataloger/Energisparepr er_og_kompaktrr/e21_DULUX_EL_teknisk_guide.pdf

[26] http://www.jwgrundy.co.uk/home_lighting.htm

[27] http://www.engineeringtoolbox.com/light-level-rooms-d_708.html

[28] “GNU Linear Programming Kit Reference Manual”, Version 4.38, May 2009.