Project for scheduling in smart grids

Arun Srikantha, Divya Pathak, Harsha Panjwani, Varun Pillai

a Department of Chemical and Biological Engineering, Drexel University

b Department of Electrical and Computer Engineering, Drexel University

c Lebow Business School, Drexel University

Case Study

An optimal energy scheduling policy that considers the varying demands in a particular residential neighborhood as well as minimizes the total cost is necessary in the present situation where global energy resources are limited. In this proposal an optimal strategy to schedule the running time of home appliances in a neighborhood is proposed. The random variations in the start time and running time of home appliances and the cost of delay incurred in case an equipment is scheduled for a later time are taken into account. The optimization formulation for scheduling the appliances in a single household is discussed in detail and the results are presented. Various practical constraints that an actual EMC (Electricity Management Controller) encounters during scheduling of appliances are discussed and the procedures to take this into account in the optimization formulations are described.

The Problem is briefly discussed here .The Scheduling problem is divided into 4 parts.

1. First is a base case problem or a simplified scenario which involves optimally scheduling the run time of three appliances for a single household.

2. The next part involves including the capacity constraints in the scheduling problem which restricts the amount of power available for a given household per time slot.

3. The last part asks to extend the optimization formulation to a neighborhood which involves communication between EMC's (Electric Management Company) of nearby homes to optimally schedule the run time of appliances.

4 Development of user interface to facilitate viewing the power consumption and appliance statistics from home user perspective and the utility company's perspective.

The Data which is used for solving the case study is as follows

Single Home

1 No of Appliances considered = Dish washer, Clothes dryer , Water heater.

2 Number of Time Periods (T) = 24 hours.

3 Cost of Electricity in $ /KWH for each time period is given in appendix.

4 Cost of Delay CD , Cn1 for time period <= 6 = $ 0.1.

5. Cost of Delay CD, Cn2 for time period >= 6 = $ 2.5.

6. The Probability data of an equipment being on /off in a given time period is given in the appendix.

7. Power Consumption for each appliance is 1.8 KWH.

Neighbourhood

1No of Homes in Neighborhood = 15.

2No of Appliances in each home = 3.

3Number of Time periods (T) = 48 hours.

4Total No of Appliances considered = 45.

5Power Consumption for each appliance in a given time period is given in Appendix.

6 Electric Cost for each appliance in a given time period is given in Appendix.

7 Cost of Delay for each time period C_n1 is as given in appendix.

8 Cost of Delay for each time period C_n2 is as given in appendix.

9 Time Periods to be considered for C_n1 is as given in appendix.

Managerial Report

Executive Summary

Energy demands in a home or across multiple homes in a neighborhood are flexible. This flexibility arises from the immediate needs of the customer to utilize various electrical appliances during the day. Primarily, electrical appliances can be categorized into two based on their utility, viz. Type I - Appliances which are required immediately for e.g. light, and Type II - Appliances which don't have to be available the minute they are turned on for e.g. dish washer, car charger etc.

Electricity is an expensive commodity. Additionally, storing electrical power is very expensive as well. Utility companies have always faced this difficulty to maintain balance between production and utilization of electricity due to the flexible nature of utilization and the high cost of supply. The flexible nature of demand for electricity causes in grid instability and results in equipment overload, brownouts & blackouts.

This issue has led to the creation of Smart Grid technologies to be applied to the electric grid for improved transmission, distribution and customer based systems. In order to analyze the customer's electricity utilization pattern as well as reducing the costs incurred for the flexible demands is through Smart Meters. These meters provide real-time electricity prices to homes.

Electricity prices are dictated by the demand for power during the day. Hence essentially, hours of the day which require high energy requirements in the grid are charged more. This information is also shared with the homes in each utility bill. Since this information is not shared real-time, the homes are unable to use it effectively.

Thus, our task was to create an Electricity Management Controller (EMC) which utilizes this information and effectively schedule the Type II appliances. The EMC would help plan and schedule Type II appliances and help in reducing customer's energy costs by shifting the load to off-peak hours.

This process thus helps balance the load to maintain grid stability as well as helps in reducing energy costs of homes. Utilizing our optimizing algorithm in the EMC, a single home benefited by saving $1.16/day.

For the utility company, the EMC helps in eliminating the energy utilization peaks which were a root cause of grid instability. The EMC achieves savings for the homes while maintaining 100% service levels. The EMC thus helps the utility company not only schedule the production/supply of electricity but also in reducing expenses due to equipment overloads and blackouts.

It also improves customer satisfaction levels by enabling the utility company to provide better service.

Electricity consumption in the Unites states is projected to increase by 29% from 2012 to 2040 [*EIA*, 2015a]. The average US residential electricity prices increased at the highest rate of 3.1% since 2008 and is projected to increase by 1.0% in 2015 and by 1.8% in 2016 [*EIA*, 2015b]. It is imperative that the utility company optimizes its processes in order to meet the residential demands of a neighborhood as well as minimize the cost of electric power consumption.

Implementation of Smart grid technologies to optimize transmission & distribution in a grid is an ongoing organization strategy.

Baseline U.S. Smart Grid Spending 2008-2017 (Historical and Forecast)

Source: BNEF 2014

We can see from the graph that major part of the spending on the Smart Grid technologies were into Smart Metering. This has resulted in installations of over 46million smart meters in the United States. Additionally, this number is estimated to reach around 65million by 2015. This accounts for over a third of the 145million meters of all types that are in use today in the United States.

A utility company's primary source of income are from the homes and offices that exist in the grid. Although these smart grid technologies help in improving the utility company's processes, it has still failed to include the end user as a part of its optimization process. This has triggered an interest to develop efficient strategies that minimizes the cost of electric power consumption by appropriately scheduling the run time of home appliances.

With these objectives in mind, we were tasked with the creation of an Electricity Management Controller (EMC) which utilizes information from the Smart Meters to effectively schedule the Type II appliances.

Primary Objectives of the EMC

- Schedule Type II Appliances in homes
- Manage power consumption and load balancing in a neighborhood

Data Considerations

The data provided to us contained probabilities of the utilization periods of three appliances in a single home. These appliances are primarily Type II appliances which can be scheduled and aren't required immediately by the home owner.

We also have real-time prices of electricity obtained from Smart Meters. This basically lists price of electricity per hour. This pricing is based on past data where peak hours were identified and the same were priced higher than the off-peak hours. The price of electricity thus remains high from the 11th to the 22nd hour period while it remains low for the rest of the periods.

Our task was to analyze the customer's electricity utilization pattern from this data and determine the costs incurred by the homeowner as a result of adhoc access of the grid.

We found that a homeowner uses 3.6 kWh /hour on an average. Additionally, this information was simulated for an entire neighborhood of 15 homes. We found that a neighborhood used 103 kWh on an average.

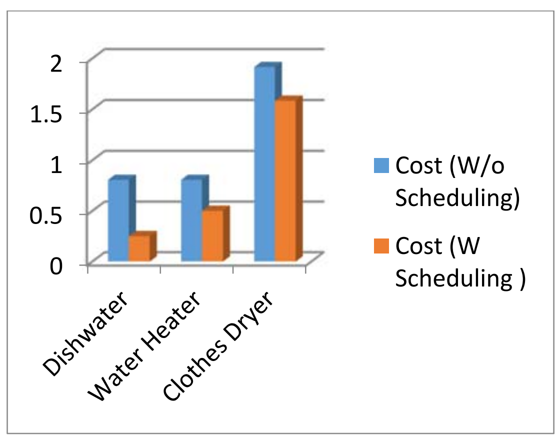
We also found examples where the grid became unstable during the simulation which proves that our simulation was a perfect sample of the real-time scenario. We ignored scenarios which did not produce any tangible benefits .

Our focus was primarily on analyzing scenarios which would provide benefits both to the customer as well the utility company.

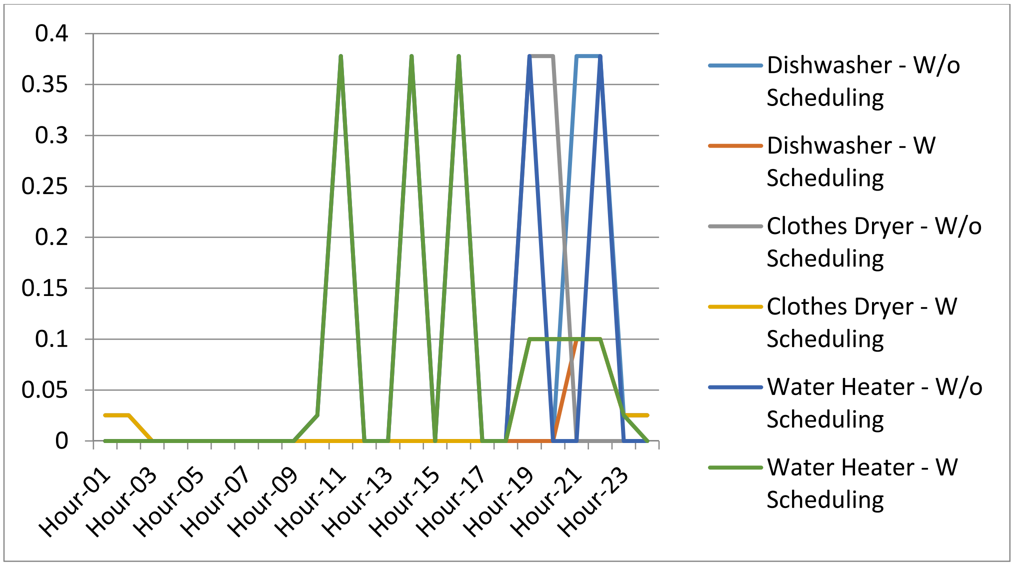
Cost Benefit Scenarios

Figure below shows the cost benefits in scheduling the appliances for a 24 hours period by applying delay costs at peak hours.

	Cost (W/o Scheduling)	Cost (W Scheduling)
Dishwasher	0.8064	0.2504
Water Heater	0.8064	0.5008
Clothes Dryer	1.9152	1.6096
Total Cost	**3.528**	**2.3608**



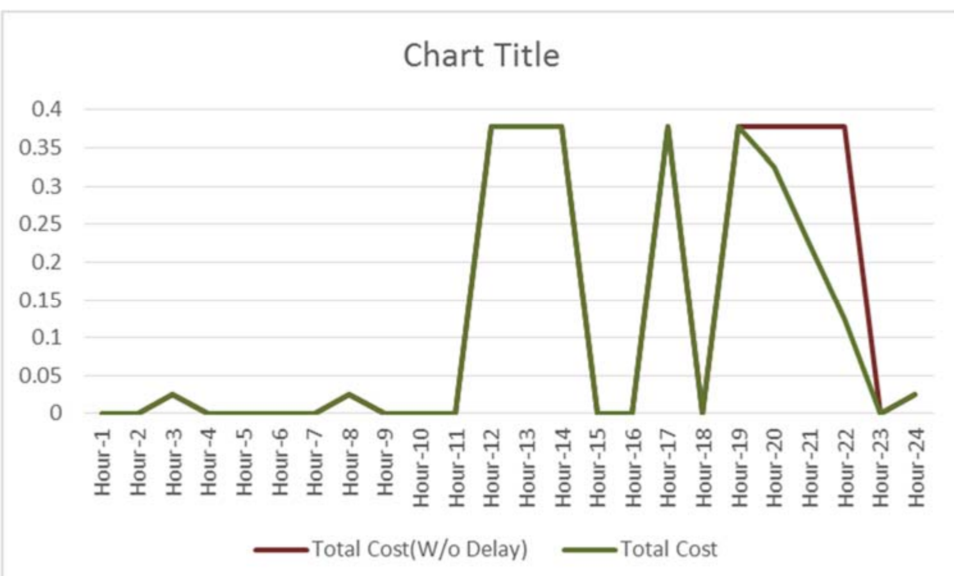
We get optimum cost benefits when delay cost is applied for a request in the 20th hour and the appliance is scheduled in the 23rd hour with the delay cost for 3 time periods.



The above graph shows the customer request for the Dishwasher in the 21st hour for a duration of 2 hours. However, the EMC scheduled to start it at the 23rd hour. Similarly, the water heater was requested at the 19th hour and 22nd hour for a duration of one hour and the clothes dryer was requested at the 19th hour for 2 hours. The EMC scheduled the clothes dryer to run at the 23rd hour instead.

EMC scheduling for the three appliances towards off-peak hours generated savings of $1.17/day for a single home.

The graph below explains the benefit we receive when we apply delay cost to schedule appliances towards non-peak hours. The ideal time period to reschedule the appliance from the simulations for a single home with delay costs applied would be for requests between 20th and 22nd hour with a maximum delay of 3 hours.



Similar to the single home scheduling, the neighborhood data was analyzed. This data considered 15homes. We ran simulations based on this data for the neighborhood and found that the appliances could be categorized further based on the duration it runs. This enables us to prioritize appliances and schedule appliances which run longer towards end of the 48 hour period we consider in the neighborhood scheduling.

Neighborhood scheduling helps maintain a level load across the power grid.

For our utility company, this has resulted in eliminating cases where the grid became unstable. This helps in eliminating the expenses caused due to equipment overload.

Appendix C

Electrical appliance scheduling for a residential neighborhood

A neighborhood of fifteen households with three appliances each is given. Each home is equipped with an EMC which schedules the run time of the appliances. The EMCs can communicate with each other and borrow power at run time. The schematic of the neighborhood problem is shown in Fig. 1.

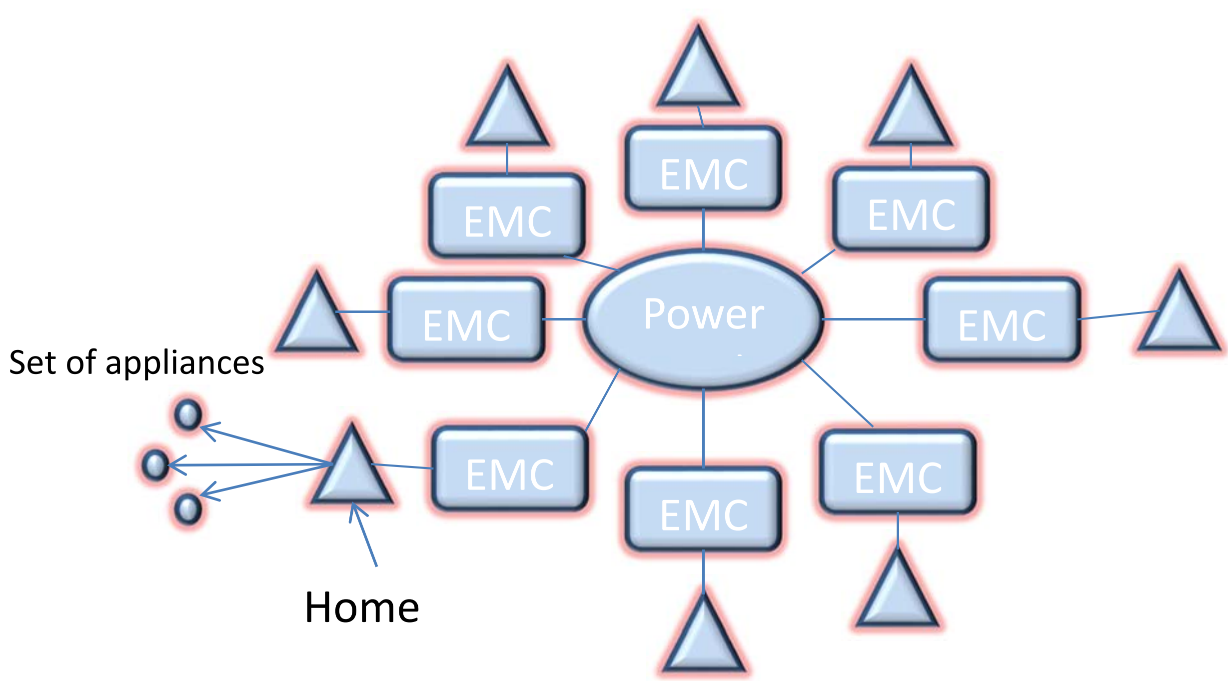


Fig 1: Schematic of residential neighborhood with 15 homes

1. Problem

Schedule the 45 appliances in a 48 hour time period such that

1. The total power consumption is below a certain threshold (Pmax) set by the Electric Company.
2. The electricity cost is optimal.

3. Homes demanding more than the maximum power limit set per home (Pmax/15) compete for extra power amongst each other as long as total power consumed is below Pmax.

2. Data provided

b_{nt}: probability that, if an appliance is 'on' at period t¸ it will be off in period 't+1'
a_{nt}: probability that, if an appliance is 'off' at period t¸ it will be on in period 't+1'
p_n: amount of power consumed by appliance n
m_n: maximum delay time allowed for appliance n
c_{n1}: cost for each period of delay within m_n periods
c_{n1}: cost for each period of delay beyond m_n periods
e_t: cost of electricity in time period t
T: length of the planning horizon (48 hours)

The values of a_{nt}, $b_{nt,}$ p_n and e_t for home 1 are shown in Figures 2 through 5 respectively.

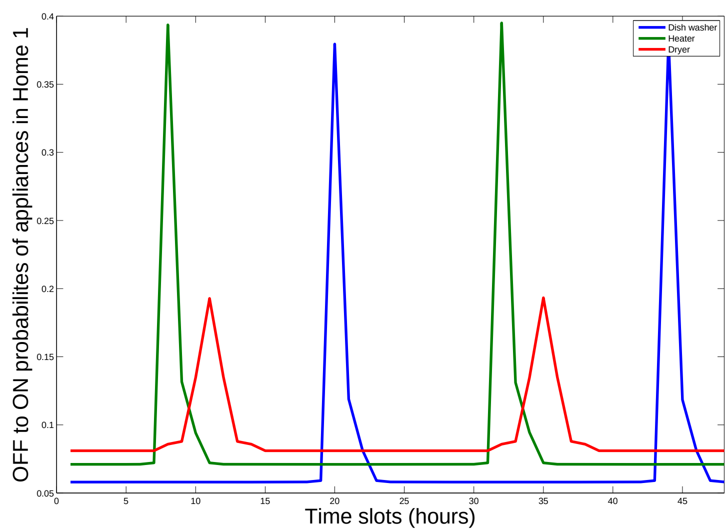


Fig 2: Appliance Off to On probabilities for Home 1

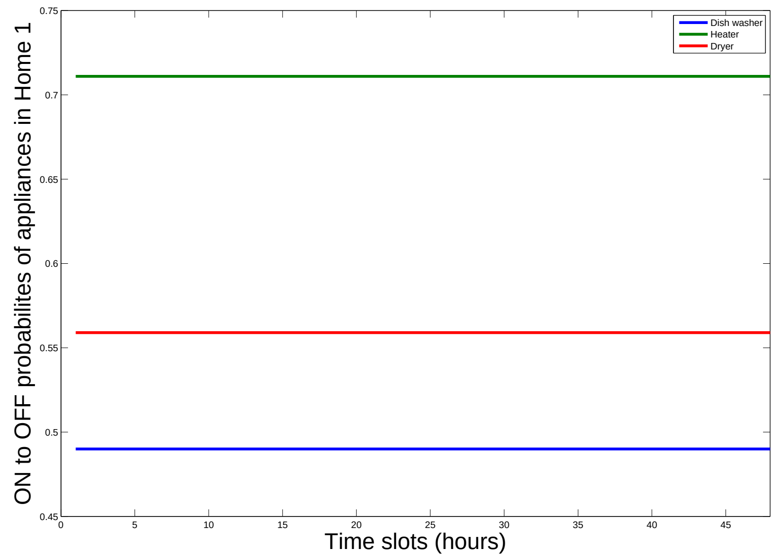


Fig 3: Appliance On to Off probabilities for Home 1

Fig 4: Power consumption of appliances in 15 homes

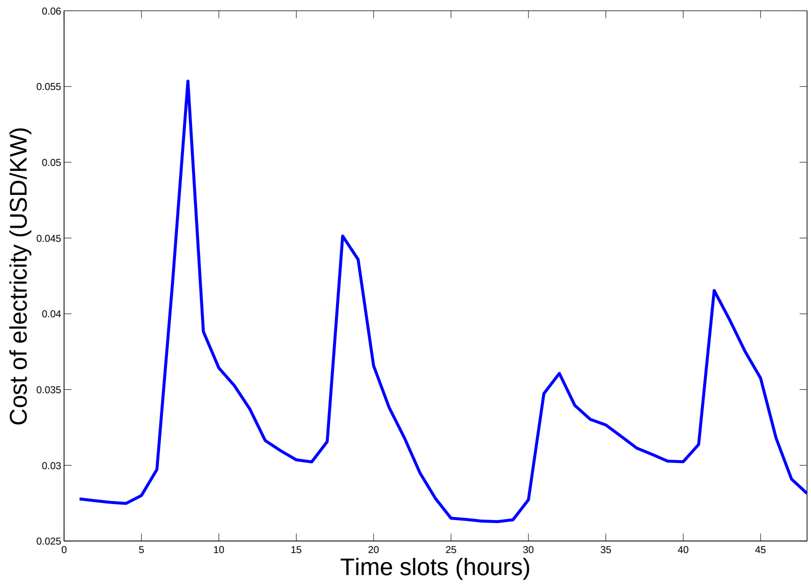


Fig 5: Cost of electricity per time period

3. Mathematical formulation of the problem

3.1 Determine appliance ON request and run time (duration)

The appliance ON request time and duration are calculated using the same technique as the base case (random number generation with value less than a_{nt}). It is assumed that the appliance is turned ON only once in the planning horizon of 48 hours. The run time of each appliance is considered as a task to be scheduled on a processor. The processor is the EMC. As all EMCs are identical, the problem reduces to scheduling N tasks on 15 processors. Each task can be represented pictorially as Fig. 6 [].

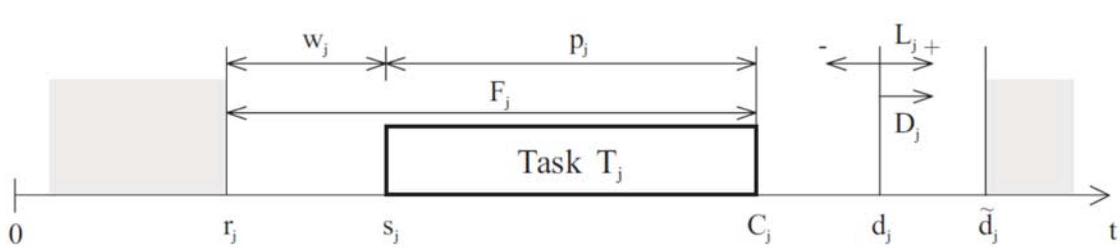


Fig 6: Graphical representation of appliance run time (task) on a single processor (EMC)

The mathematical terms in Fig. 6 are described below:

r_j : Appliance request time. It is the time at which a task becomes ready for execution. This is obtained through the random number generation per time slot.
p_j (duration): Appliance run time or duration. This is the time necessary to execute task Tj on the processor without interruption (preemption is not allowed).
d_j : The time limit by which the task has to be completed, otherwise the scheduling is assumed
to fail. This is a hard deadline and for the given problem, it is set to infinity because once an ON request is granted, the appliance can run forever.
$\tilde{d}_j$: The time limit by which the task should be completed. This is the due date for completion of the tasks and is set to the end of the planning horizon (48 hours).
s_j : The task starts time. This is obtained from the scheduler.

3.2 Problem definition

Minimize total electricity cost:

$$C = max\ (0,\ (duration - m_n)\ C_{n2}) + min\ (duration,\ m_n)\ C_{n2}$$

Subject to the total power constraint:

$$\sum duration * p_n \leq P_{max} \qquad (1)$$

(Sum of all running appliances in the planning horizon)

3.3 Scheduling Algorithm

Multiple scheduling algorithms from the TORSCHE [11] scheduling toolbox in MATLAB were analyzed. Two algorithms were found suitable to solve the problem: algorithm for problem 1|rj|Cmax which is similar to first come first serve and earliest starting time first (EST). The description of the algorithms is available in [11]. The usage of the scheduling algorithm is given in section 4 and the flowchart is shown in Fig. 7.

It is observed that there is no significant variation in the electric power consumption cost including the delay penalty. This is unlike the base case where the cost of power consumption is significantly high in certain time periods. As seen in Fig. 5, the maximum power consumption cost variation is USD 0.03. Therefore, the appliance scheduling is optimized to meet the power constraint in (1).

[11] Torsche scheduling toolbox for MATLAB, “http://rtime.felk.cvut.cz/scheduling-toolbox/”

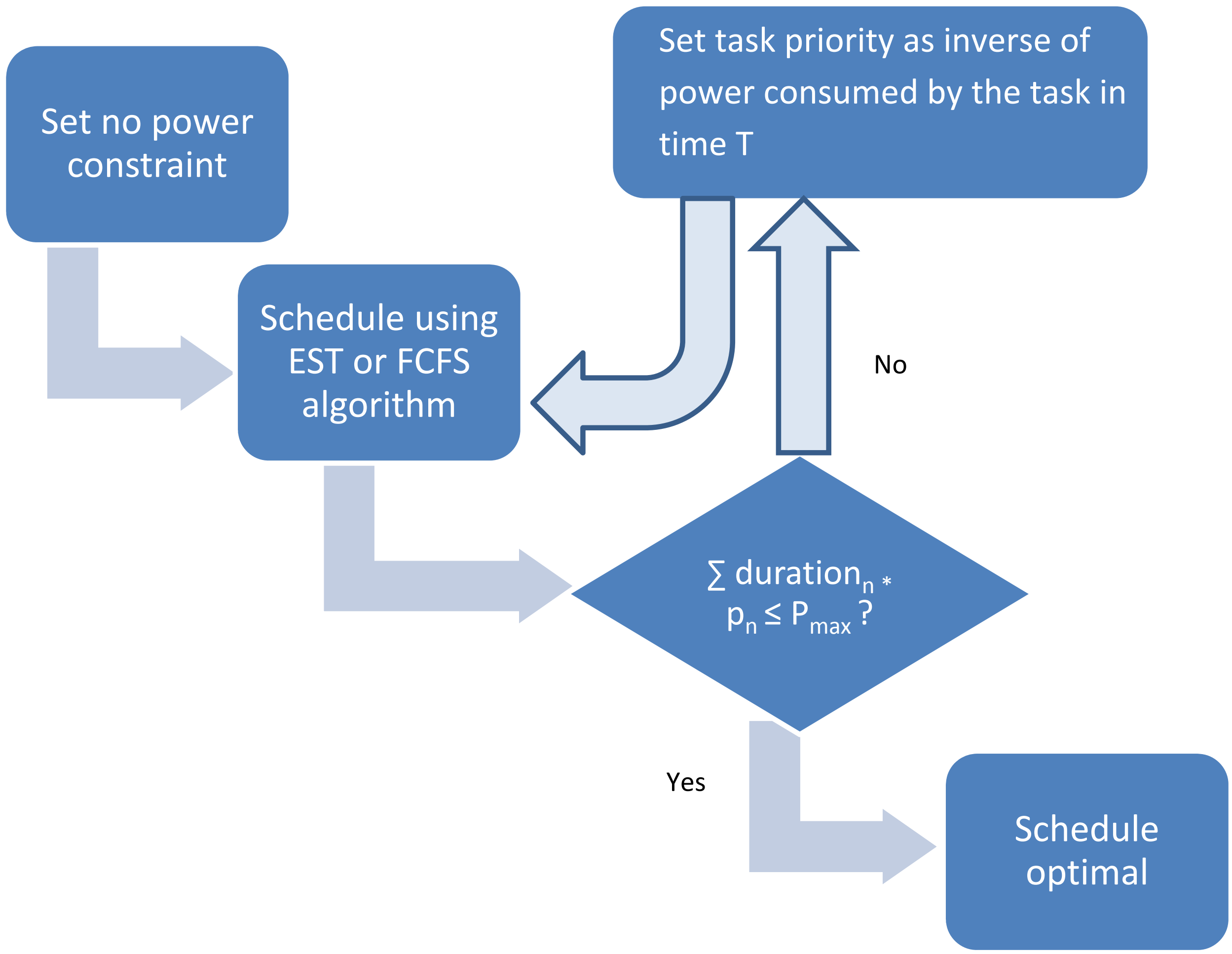


Fig. 7: Appliance scheduling flow chart

The appliance scheduling obtained from the algorithm is shown as a Gantt chart in Fig. 8.

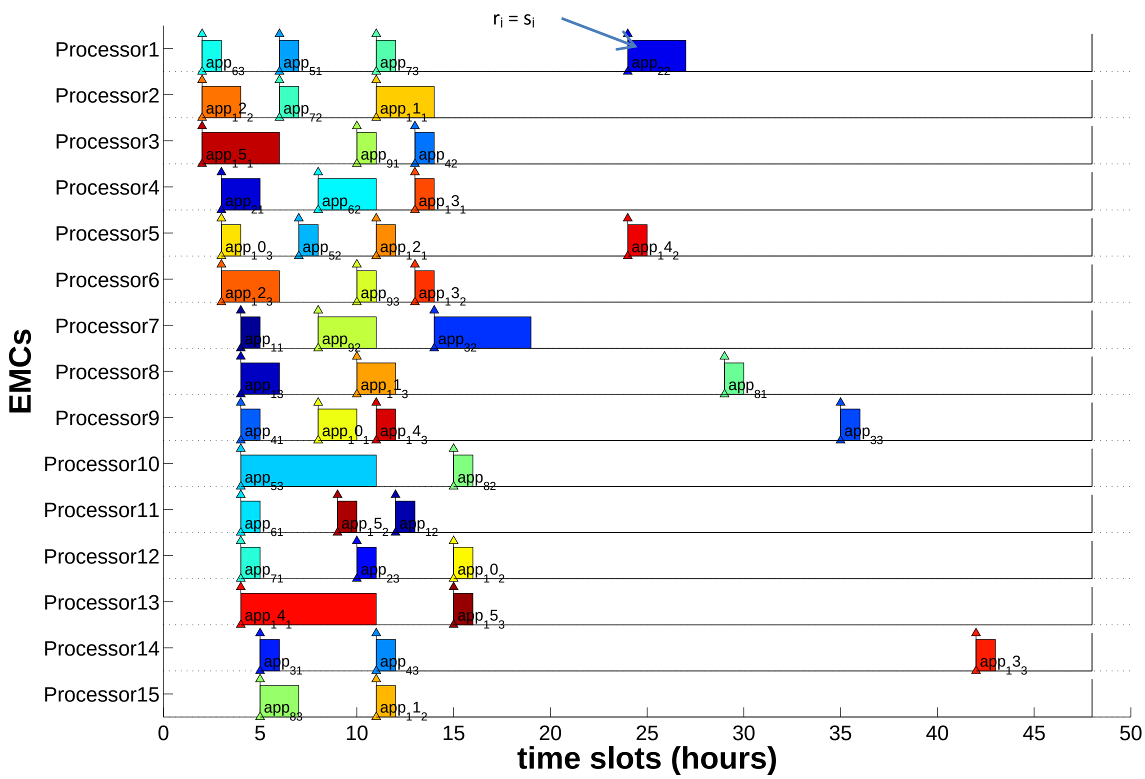


Fig 8: Appliance scheduling on 15 EMCs (parallel processors) using EST algorithm

3.4 Determination of P_{max}

From the given data set, an optimum value of P_{max} is determined such that all the appliances requesting to be turned ON in the 48 hour time period are granted access. This ensures a quality of service (Qu's) level of 100%. The appliances were scheduled using the EST algorithm for 1000 randomly generated scenarios of appliance run time and duration. Table 1 provides the total power consumption statistics. P_{max} is thus set to the mean value of the power consumed by the 45 appliances (103 KW).

Mean (KW)	Mode (KW)	Median (KW)	Max (KW)
102.97	94.9	104.7	126.9

Table 1: Statistical data of total power consumed by appliances running in 48 hour time period

3.5 Trade-off analysis between P_{max} and Qu's

To obtain further power savings for the electric company and the neighborhood association, the value of P_{max} is lowered below 103 KW. The appliances are now scheduled with a modified version of FCFS algorithm. A priority is assigned to each task which is inversely proportional to the power consumed by the task. This is a greedy heuristic which schedules all the tasks which arrive early and have least processing time. Lowering P_{max} below 70 KW requires more than one iteration of the algorithm shown in Fig 7 and the Qu's degrades to 80%. This is shown pictorially in Fig. 9. 15 appliances crossed the due date of 48 hours.

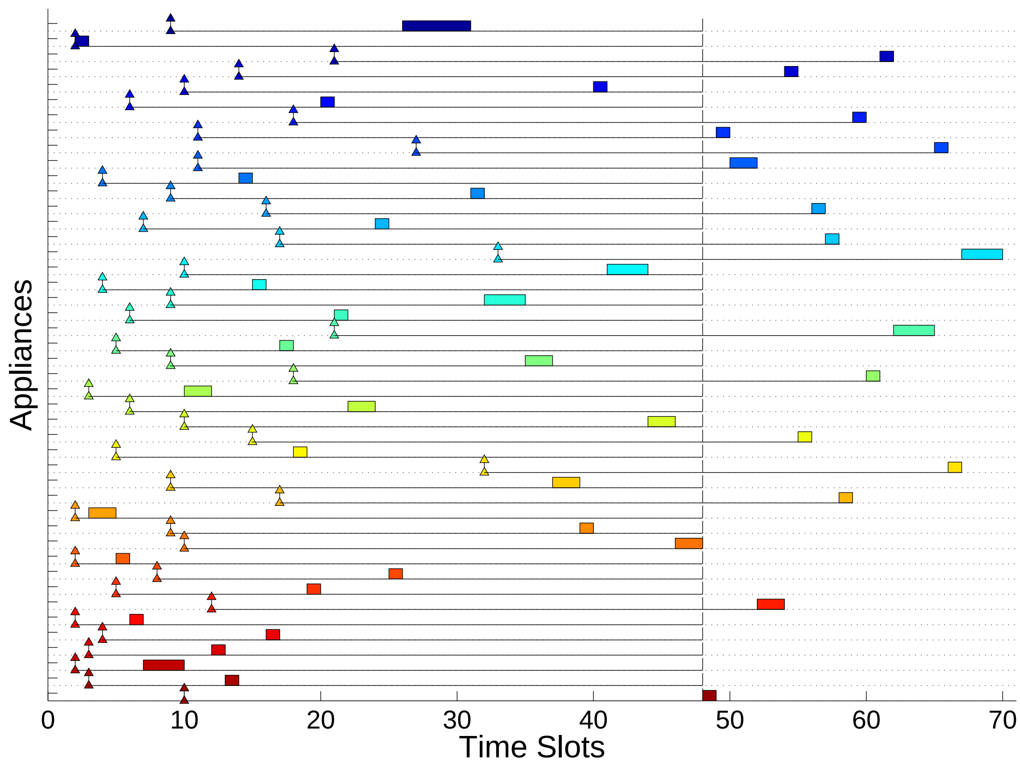


Fig 9: Appliance scheduling by assigning weights to the tasks as per their power consumption.

4. Matlab code for the neighborhood appliance scheduling with power constraint

Declare variables and pre-allocate matrices

```
num_house = 15;
% Number of appliances
num_app = 45;
% Total time slots of length one hour
time_slots = 48;
% Appliance stop time
stop_time = zeros(1,num_app);
% Appliance run time
duration = zeros(1,num_app);
% Appliance delay cost
Cd = zeros(1,num_app);
% Appliance power consumption cost when ON request granted immediately
Cpr = zeros(1,num_app);
% Appliance scheduling task
app_task = zeros(1,num_app);
% Power consumed per appliance
pow_app = zeros(1,num_app);
% Power limit per home (kW) per time slot of one hour
% Obtained through sensitivity analysis done on neigborhood level max power
% consumption
Pmax_home = 7; %Not used currently
% Power limit per neigborhood (kW) per day (24 time slots)
Pmax_neighbor = 105; % This the average value obtained from the data given
```

Data for one neighborhood with 15 households

Static data, per appliance

```
load m_n;
load c_n1;
load c_n2;
load p_n;

% Dynamic data per time slot
% The electricity cost is almost same for all 48 hours,
% thus schedule the appliance as soon as requested
load e_t;
load a_nt;
load b_nt;
n = 1;
```

```
j = 1;



% split up the given probability data into matrics (time_slotsX num_app)
for k = 1:num_app
   for i = n:n+time_slots-1
       a_t(i-n+1,k) = a_nt(i);
       b_t(i-n+1,k) = b_nt(i);

   end
   n = n+time_slots;
end

% name string for appliances: House#_app#
app_name = cell(45,1);
n = 1;
for i = 1:num_house
   for j = 1:3
       house = int2str(i);
       app = int2str(j);
       app_name{n} = strcat('app_',house,'_', app);
       n = n+1;
   end
end
app_name = app_name';
```

Generate random scenarios for appliance ON time and OFF time

Assumption: Consider only the first instance of appliance turned ON in the 48 hour period

```
Ton = zeros(time_slots,num_app);
Toff = ones(time_slots,num_app);

for j = 1:num_app
   for i= 1:time_slots-1
     if (Toff(i,j))
         if(rand(1,1) < a_t(i,j))
              Ton(i+1,j) = 1;
              Toff(i+1:24,j) = 0;
              break;
         end
     end
   end
```

```
end
% request_time = index of first 1 in Ton matrix
[C, request_time] = max(Ton);
% stop_time = index of first 1 in OFF array
for j = 1:num_app
  for i= request_time(1,j):time_slots
    if(rand(1,1) < b_t(i,j))
      stop_time(1,j) = i+1;
      break;
    end
  end
end
```

Cost formulation

```
for i = 1:num_app
  % Obtain the equipment ON duration
  duration(i) = stop_time(i) - request_time(i);
  %Power Consumption cost if appliance is not delayed (no scheduling)
  Cpr(i) = sum(e_t(request_time(i):stop_time(i)));
end
% Total cost per house
for i = 1:num_house
  cost_house_initial(i) = Cpr(1,i)+Cpr(1,i+15)+Cpr(1,i+30);
end
```

Schedule each appliance to obtain feasible start times

Step 1: No power constraint per home in scheduling the appliances

A hard deadline of inf is set for the completion of run-time of each appliances A due date equal to 48 hours is set for all appliances

```
for j = 1:num_app
  app1((j)) = task('',duration(1,j), request_time(1,j), inf, time_slots);
end
%app_task1 = taskset(app1());
app_task1 =
taskset([app1(1,1),app1(1,2),app1(1,3),app1(1,4),app1(1,5),app1(1,6),app1(1,7),app1(1,8),app1(1,9),app1(1,10),app1(1,11),app1(1,12),app1(1,13
),app1(1,14),app1(1,15),app1(1,16),app1(1,17),app1(1,18),app1(1,19),app1(1,20),app1(1,21),app1(1,22),app1(1,23),app1(1,24),app1(1,25),app1(
1,26),app1(1,27),app1(1,28),app1(1,29),app1(1,30),app1(1,31),app1(1,32),app1(1,33),app1(1,34),app1(1,35),app1(1,36),
app1(1,37),app1(1,38),app1(1,39),app1(1,40),app1(1,41),app1(1,42),app1(1,43),app1(1,44),app1(1,45)]);
% Two scheduling algorithms are tried for optimum scheduling:
% Earliest Starting Time first (EST)
```

```
% First Come First Serve (FCFS)
% EST offers the advantage of parallelizing the scheduled tasks on
% available processors (EMCs). Since all EMCs are identical and can share
% power with maximum power limit set per neigborhood and not per EMC, EST
% provides the optimal solution
prob = problem('P|rj|sumCj');
% Assign all appliances to 15 EMCs (processors)
TS = listsch(app_task1,prob,15,'EST');
figure();
plot(TS);
% schedule the appliances on first come first serve basis
prob1=problem('1|rj|Cmax');
TS1=alg1rjcmax(app_task1,prob1);
figure();
plot(TS1,'proc',0);
% Get the scheduled start times for all appliances
[start_time, length, processor, is_schedule] = get_schedule(TS);
% Delay time for each appliance
delay = start_time - request_time;

% Number of appliances that could complete their runtime in 48 hours
num_scheduled = 0;
for i = 1:num_app
    if (start_time(i) + duration(i) < time_slots)
      pow_app(i) = p_n(i)*duration(i);
      % Assign priority to appliances based on their total power consumption
      % as per original schedule
      priority(i) = 1/pow_app(i);
      num_scheduled = num_scheduled+1;
    else
      % Appliances which did not start in 48 hour period or did not
      % complete are assigned least priority
      priority(i) = 0;
    end
end
% Total power consumed till run time of all appliances
total_power = sum(pow_app);
% Since obtained priorities are fractions
priority = ceil(tiedrank(priority));
```

Utility Perspective

Schedule appliances such that QoS is not impacted while the power constraint is met Calculate the power consumption for all scheduled appliances in 48 hours

Reschedule appliances if total power consumed in original schedule exceeds neighborhood level power limit (Trade off analysis)

```
if (total_power > Pmax_neighbor)
   for j = 1:num_app
     app2((j)) = task('',duration(1,j), request_time(1,j), inf, time_slots, priority(1,j));
   end
   app_task2 =
taskset([app2(1,1),app2(1,2),app2(1,3),app2(1,4),app2(1,5),app2(1,6),app2(1,7),app2(1,8),app2(1,9),app2(1,10),app2(1,11),app2(1,12),app2(1,13
),app2(1,14),app2(1,15),app2(1,16),app2(1,17),app2(1,18),app2(1,19),app2(1,20),app2(1,21),app2(1,22),app2(1,23),app2(1,24),app2(1,25),app2(
1,26),app2(1,27),app2(1,28),app2(1,29),app2(1,30),app2(1,31),app2(1,32),app2(1,33),app2(1,34),app2(1,35),app2(1,36),
app2(1,37),app2(1,38),app2(1,39),app2(1,40),app2(1,41),app2(1,42),app2(1,43),app2(1,44),app2(1,45)]);
% Modified the first come first serve algorithm to consider scheduling along with assigned priorities while scheduling the tasks
   prob2=problem('1|rj|Cmax');
   TS2=alg1wjcmax(app_task2,prob2,15);
   [start_time2, length, processor2, is_schedule] = get_schedule(TS2);
   figure();
   plot(TS2,'proc',0);
   % Calculate the power consumption for all scheduled appliances in 48 hours
   % Number of appliances that could complete their runtime in 48 hours
   num_scheduled2 = 0;
   for i = 1:num_app
      if (start_time2(1,i) + duration(i) < time_slots)
         pow_app2(i) = p_n(i)*duration(i);
         num_scheduled2 = num_scheduled2+1;
      end
   end

% Total power consumed in 48 hour period
total_power2 = sum(pow_app2);
```

For home user, calculate the cost of power consumed from the start time.

For cost optimization: Rerun scheduling by assigning weights as per cost

Calculate the cost of power consumed for all scheduled appliances Delay time for each appliance

```
delay = start_time2 - request_time;
for i = 1:num_app
    if (start_time2(i) < time_slots)
        if (start_time2(i)+duration(i) < time_slots)
           Cp(i) = sum(e_t(start_time2(i):(duration(i)+start_time2(i))));
        else
           Cp(i) = sum(e_t(start_time2(i):time_slots));
```

```
        end
        Cd(i) = max(0,(delay(i)-m_n(i)))*c_n2(i) + min(delay(i),m_n(i))*c_n1(i);
    end
end
% Total cost per appliance
C = Cp + Cd;
% Total cost per house
for i = 1:num_house
   cost_house(i) = C(1,i)+C(1,i+15)+C(1,i+30);
end

% Cost saving per appliance
C_save = Cpr - C;
%Cost savings per house
c_house_save = cost_house_initial-cost_house;
% Power savings
pow_sav = total_power - total_power2;

end
```

2 Motivation

The time at which an appliance is used and the time duration for which it is used in a day is often flexible. This is advantageous for an EMC (Electrical Maintanence Company), since an appliance if requested during a time slot when the cost of electric consumption is high, can be rescheduled to a later time to reduce the cost, since the cost of electric power may be low at a later time. However, such resheduling may result in an additional delay cost associated with an appliance which EMC has to entail. Hence a tradeoff exists between delaying the run time of an appliance and the cost associated with delaying that appliance. This tradeoff can then be utilized to reduce the overall cost of electric power by appropriately scheduling the appliances. In this work, the problem of scheduling appliances is considered for three scenarios and an optimization model is developed. The three scenarios are,

- Base case (Fig. 1): Scenario consisting of a single household with three appliances.
- Basecase with capacity constraints: Single household restricted by power usage per time period.
- Scheduling appliances for neighbourhood consisting of individual homes.

For the analysis, the data provided in 4th AIMMS-MOPTA optimization modelling competition is considered. Specifically the data provided in "smalldata.xls" is considered in this work. The description of the data and a preliminary analysis of the data is explained in the next section.

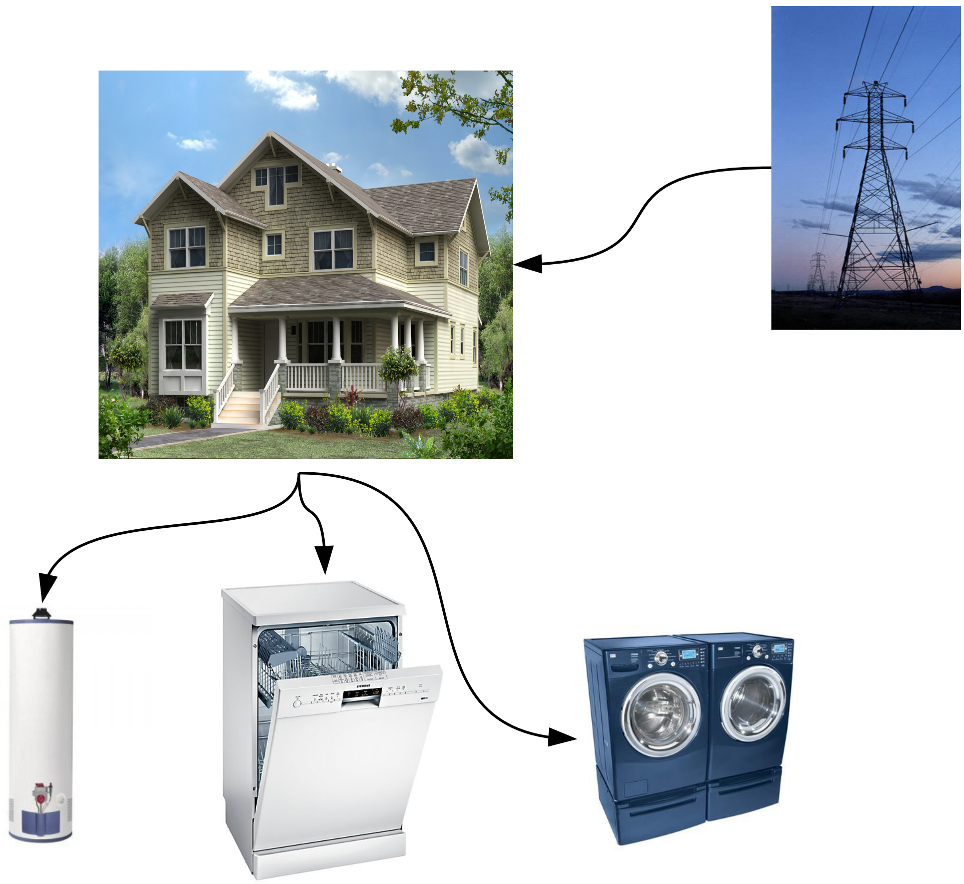


Figure 1: A schematic representation of base case scenario

3 Data generation and analysis

The "smalldata.xls" contains the data obtained for single household with N_a appliances where $N_a = 3$, representing a dish washer, clothes dryer and a water heater. Further the data collected is such that, a day is split into n equal time where each time slot is denoted by t. If an appliance is off in period t, then the probability that it is requested in period $t+1$ is a_{nt} . Similarly, if an appliance is on in period t, then the probability that it is off in period $t+1$ is b_{nt}. Based on this probability data for each appliance and by using random number generator from a uniform distribution, the request time slots and the duration time slots (the time slots upto which an appliance is in use) for the appliances used in a day are generated. This procedure is schematically represented in Fig.2 for an appliance. A Frequency distribution of the timeslots and duration slots for the three appliances are generated by considering 10000 samples i.e., appliance usage per day over a period of 10000 days. Further the sampling is restricted to appliance usage that is not beyond 23 hours (samples for which the duration slots exceeding 23 are not allowed). A request slot of zero implies that the appliance is not used in that day. The frequency distribution provides useful insights about an appliance usage which are helpful in two aspects.

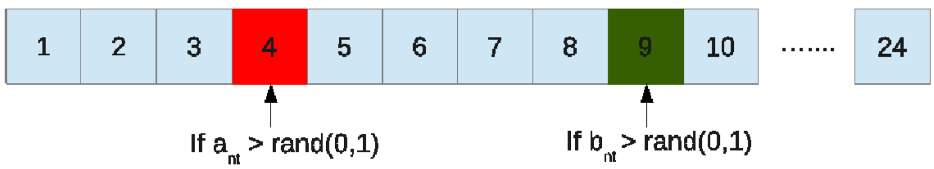


Figure 2: Generation of request time slot and duration slots for an appliance

First, the distribution assists in performing importance sampling. For eg., when no equipment is in use in a particular day, then the there is no electric power consumption. From Fig.2, it can be observed that out of 10000 days, the dish washer has been used only for about 5000 days and like wise for the clothes dryer and the water heater. Hence, these scenarios can be removed to reduce the computational burden of cost function evaluations in computing or minimizing the expected cost.

The second is for developing heuristic approaches or greedy alogorithms for scheduling. For eg., the probability that the dish washer is requested to be used is high during the $17^t h$ time slot which is a period where the cost of electric power is high (Fig. 3). Hence one can schedule the dish washer to a later time if one receives a request for the dishwaher between $17^t h$ and 22^{nd} time slots. However, such greedy alogorithms do not yield long term profit and a formal optimization model needs to be formulated and solved. The optimization is explained in the next section.

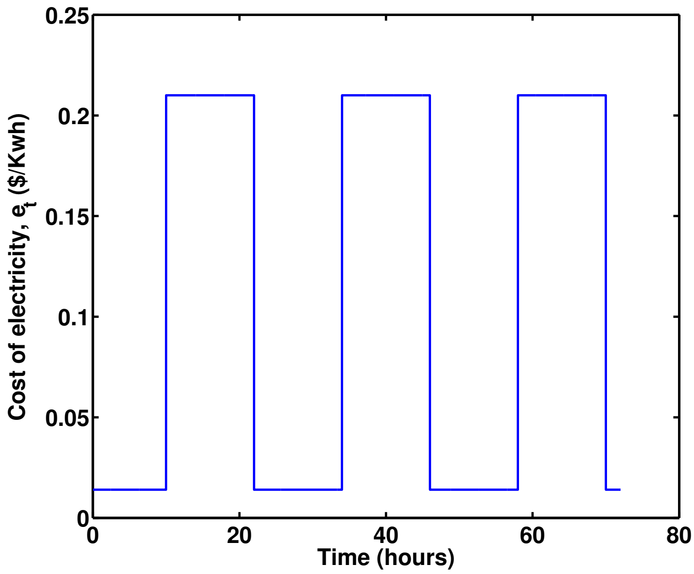


Figure 3: Costfunction

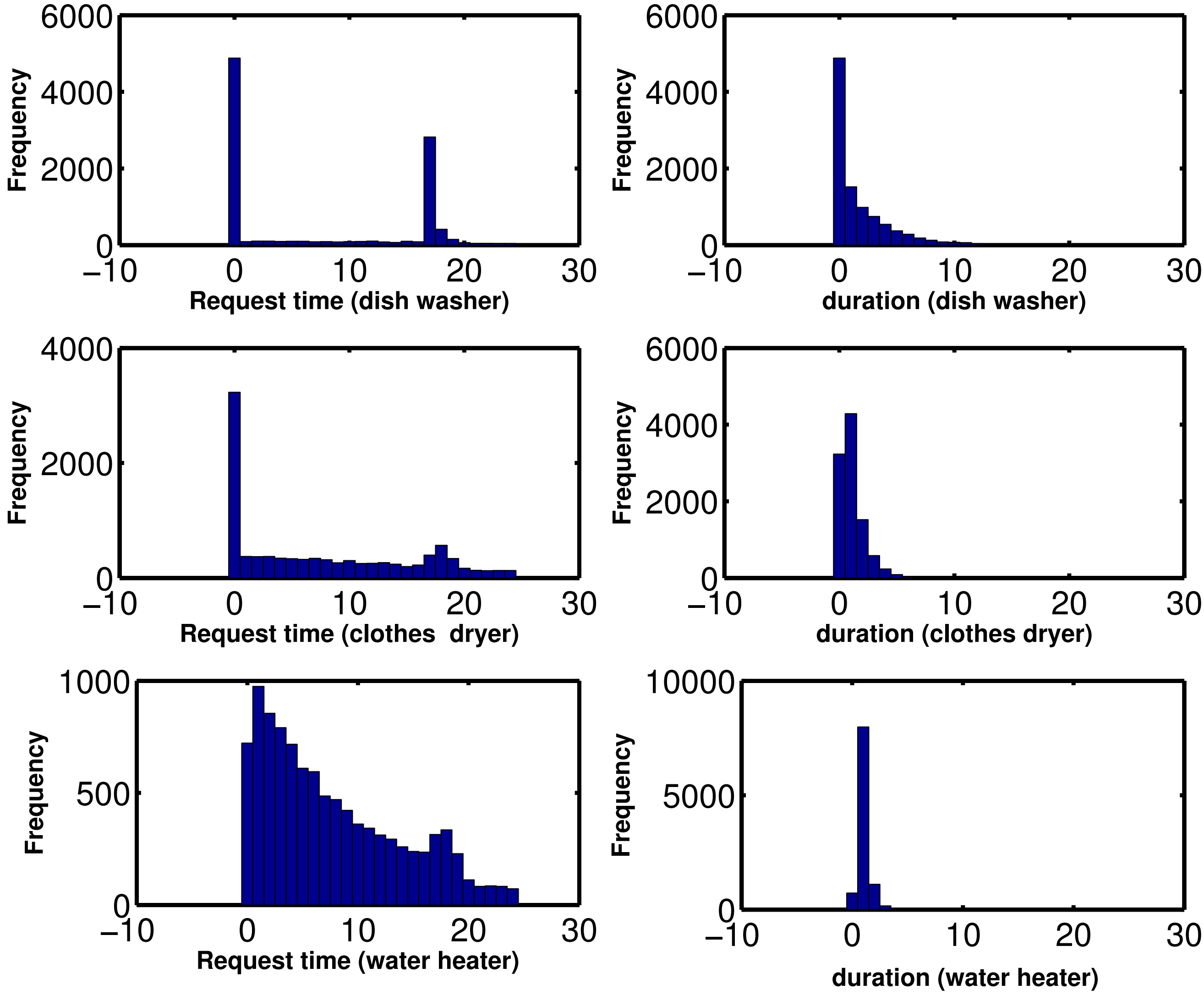


Figure 4: Frequency distribution of request time slots and the corresponding duration time slots of appliances

4 Optimization

As explained in section 2, a trade off exists between between delaying the run time of an appliance and the cost associated with delaying that appliance. In this section, a formal optimization model is then proposed to explore this trade-off. The optimization model for each of the scenarios mentioned in section 2 is then solved and the results are discussed below. The decision variables in the optimization model are the delay time periods, $\tau_j, j = 1...N_a$ associated with each appliance and are used integer variables. The various costs associated with scheduling an appliance is given below.

The cost of electric power consumption by an appliance given by,

$$c_{e,j}(\tau_j, R_{k,j} = r_{k,j}) = \sum_{i=1}^{r_k+d} e_t(\tau_j, R_{k,j} = r_{k,j})P_j \quad, j = 1, 2....N_a, \;\; k = 1, 2....N \tag{1}$$

where P_j, r are the power ($/KWh) consumed by an appliance and the request time slot respectively. The request time slot is a stochastic variable and is obtained as mentioned in section 3

The cost of delay associated with an appliance when it is delayed by τ_i time slots is then,

$$c_{d,j}(\tau_j) = c_{nj}\tau_j \;\; \tau_j \leq m_n, \;\; , j = 1...N_a \tag{2}$$

if τ_i exceeds m_n time periods, an additional cost is involved in scheduling that appliance which is given by,

$$c_{ad,j} = c_{nj}, \;\; \tau_j > m_n \;\; j = 1...N_a \tag{3}$$

The total cost of running an appliance is,

$$C_{T,j}(\tau_j, R_{k,j} = r_{k,j}) = c_{e,j} + c_{d,j} + c_{ad,j} \quad , j = 1.....N_a \tag{4}$$

The cost associated with a household is then given by,

$$C_H(\tau_{j=1..N_a}, R_{k,j=1..Na}) = \sum_{j=1}^{j=N_a} C_{T,j} \tag{5}$$

The expected cost per day is then calculated using the following approximation,

$$E(C_H) \sim \frac{C_H}{N} \tag{6}$$

The optimization model to be solved is then given by,

$$\min_{\tau_{i=1...N_a}} \;\; E\left[C_H(\tau_{j=1..N_a}, R_{k,j=1..Na} = r_{k,j=1....N_a})\right] \;\; ,..,,,,\tau_i \epsilon \mathbb{Z} \tag{7}$$

4.1 Solution methodology

The optimization problem is an NILP (Nonlinear Linear Integer Programming) problem. This is then solved using MIDACO-SOLVER [$MIDACO$, 2015], a global optimization software for MINLP (Mixed Integer Nonlinear Programming) problem. This solver utilizes ant colony optimization technique to obtain the optimal solution. The solution obtained may not be a global solution and different initial guesses are used

The optimization model is solved on a

5 Basecase scenario

The optimization model described in section 4 is solved for the base case scenario with the parameters $m_n = 6$, $N_a = 3$ and for $N = 100, 1000, 10000$ to obtain the optiaml scheduling policy. The results of the optimal solution is summarized in Table. 1.

Table 1: Optimal solutions

$E^*(C_H)$/day (\$)	N (days)	$(\tau_1^*, \tau_2^*, \tau_3^*)$
0.27354000	100	(6,3,8)
0.31673280	1000	(6,3,8)
0.30562032	10000	(6,3,8)

Since the solution obtained may not be global a consistency check was performed with different initial guesses. The optimal solutions indicates the mean scheduling policy i.e., the average of the daily scheduling should flutuate around the mean values of $\tau_1^*, \tau_2^*, \tau_3^*$ and the expected cost represents a long term average of the cost of electric power incurred every day.

6 Base case scenario with capacity constraints

In the basecase scenario it was assumed that the household has no restrictions on the power consumption. However a household is usually restricted by power consumption. Typically a house hold is restricted by the power consumption per time slot. Hence the optimization model that needs to be solved is,

$$\min_{\tau_{i=1...N_a}} \quad E\left[C_H(\tau_{j=1..N_a}, R_{k,j=1..Na} = r_{k,j=1....N_a})\right] \quad \tau_i \epsilon \mathbb{Z} \tag{8}$$

s.to

$$\sum_{l=1} P_{jl} = P_{max} \quad l = 1........ \tag{9}$$

where l represents the number of time slots in a period of N days. The P_{max} was chosen to be 3.6 which is the power required to run utmost two apliances in the time slot. To reduce the computational burden and for illustration a small sample size of 100 (100 days) is chosen.

The optimization model is then solved. For the 100 scenarios that are generated it can observed from Fig. 5 that on certain time slots the home exceeds P_{max}. From Fig. 6 it can be observed that after optimally rescheduling, the load constraints per time slot are satisfied and the optimal solution is again (6,3,8) for $(\tau_1^*, \tau_2^*, \tau_3^*)$

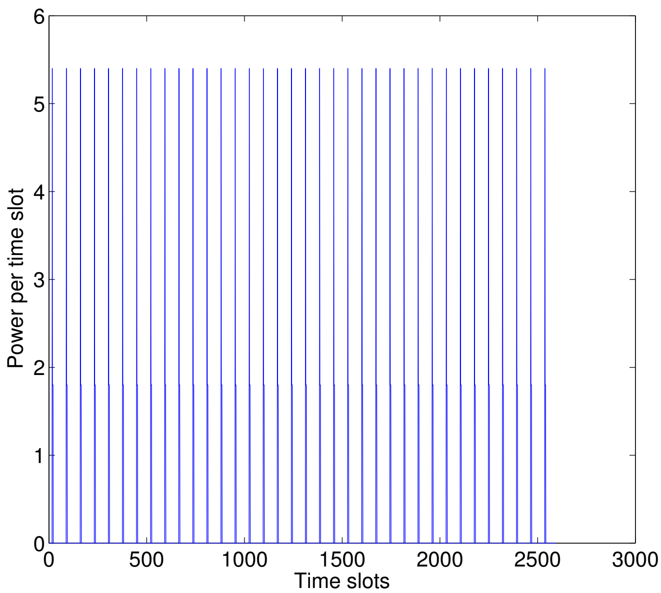


Figure 5: Load distribution in a time slot before scheduling

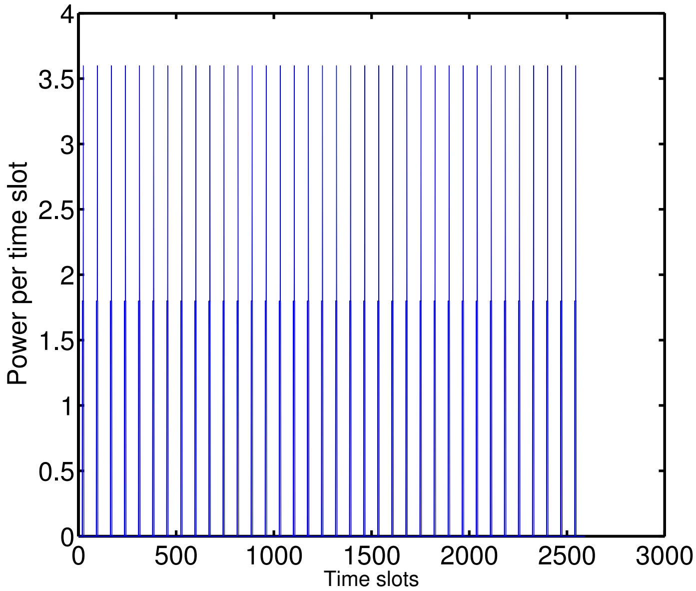


Figure 6: Load distribution in a time slot after scheduling

References

Bu, S., Yu, F., Liu, P., and Zhang, P. (2011). Distributed scheduling in smart grid communications with dynamic power demands and intermittent renewable energy resources. In *Communications Workshops (ICC), 2011 IEEE International Conference on*, pages 1–5.

Burcea, M., Hon, W.-K., Liu, H.-H., Wong, P., and Yau, D. (2013). Scheduling for electricity cost in smart grid. In Widmayer, P., Xu, Y., and Zhu, B., editors, *Combinatorial Optimization and Applications*, volume 8287 of *Lecture Notes in Computer Science*, pages 306–317. Springer International Publishing.

Caron, S. and Kesidis, G. (2010). In *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pages 391–396.

Chen, C., Kishore, S., and Snyder, L. (2011). An innovative rtp-based residential power scheduling scheme for smart grids. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 5956–5959.

EIA (accessed March, 2015a). $http://www.eia.gov/forecasts/aeo/mt_electric.cfm.$

EIA (accessed March, 2015b). $http://www.eia.gov/forecasts/aeo/mt_electric.cfm.$

Joe-Wong, C., Sen, S., Ha, S., and Chiang, M. (2012). Optimized day-ahead pricing for smart grids with device-specific scheduling flexibility. *IEEE Journal on Selected Areas in Communications*, 30(6):1075–1085.

Koutsopoulos, I. and Tassiulas, L. (2012). Optimal control policies for power demand scheduling in the smart grid. *Selected Areas in Communications, IEEE Journal on*, 30(6):1049–1060.

MIDACO (accessed March, 2015). http://www.midaco-solver.com/index.php/download/matlab.

Zhang, D., Shah, N., and Papageorgiou, L. (2013). Efficient energy consumption and operation management in a smart building with microgrid. *ENERGY CONVERSION AND MANAGEMENT*, 74:209–222.

```
 MIDACO 4.0    (www.midaco-solver.com)
 -------------------------------------

 LICENSE-KEY:  MIDACO_LIMITED_VERSION___[CREATIVE_COMMONS_BY-NC-ND_LICENSE]

 -------------------------------------
 | N      3    | MAXEVAL        4000 |
 | NI     3    | MAXTIME       86400 |
 | M      0    | PRINTEVAL       500 |
 | ME     0    | SAVE2FILE         1 |
 -------------------------------------
 | PARAM(1)  0.001000  ACCURACY G(X) |
 -------------------------------------
 | PARAM(2)     43534.0  RANDOM-SEED |
 | PARAM(3)         0.0  FSTOP       |
 | PARAM(4)       500.0  AUTOSTOP    |
 | PARAM(5)         0.0  ORACLE      |
 | PARAM(6)         0.0  FOCUS       |
 | PARAM(7)         0.0  ANTS        |
 | PARAM(8)         0.0  KERNEL      |
 | PARAM(9)         0.0  CHARACTER   |
 -------------------------------------

 [     EVAL,    TIME]        OBJECTIVE FUNCTION VALUE         VIOLATION OF G(X)
 ------------------------------------------------------------------------------
 [        1,       0]        F(X):         0.32533200         VIO:     0.000000
 [      500,      15]        F(X):         0.27354000         VIO:     0.000000
 [     1000,      30]        F(X):         0.27354000         VIO:     0.000000
 [     1500,      45]        F(X):         0.27354000         VIO:     0.000000
 [     2000,      60]        F(X):         0.27354000         VIO:     0.000000
 [     2500,      75]        F(X):         0.27354000         VIO:     0.000000
 [     3000,      91]        F(X):         0.27354000         VIO:     0.000000
 [     3500,     106]        F(X):         0.27354000         VIO:     0.000000
 [     4000,     121]        F(X):         0.27354000         VIO:     0.000000

 OPTIMIZATION FINISHED  --->  MAXEVAL REACHED


         BEST SOLUTION FOUND BY MIDACO
 --------------------------------------------
 EVAL:      4000,  TIME:     121,  IFLAG:   1
 --------------------------------------------
 F(X) =                     0.273540000000000
 --------------------------------------------        BOUNDS-PROFIL
 x(   1) =                 6.000000000000000;  %  ____________x________
 x(   2) =                 3.000000000000000;  %  ______x______________
 x(   3) =                 8.000000000000000;  %  ________________x____
```

```
MIDACO 4.0    (www.midaco-solver.com)
-------------------------------------

LICENSE-KEY:  MIDACO_LIMITED_VERSION___[CREATIVE_COMMONS_BY-NC-ND_LICENSE]

-------------------------------------
| N      3    | MAXEVAL        4000 |
| NI     3    | MAXTIME       86400 |
| M      0    | PRINTEVAL       500 |
| ME     0    | SAVE2FILE         1 |
-------------------------------------
| PARAM(1)  0.001000  ACCURACY G(X) |
-------------------------------------
| PARAM(2)     43534.0  RANDOM-SEED |
| PARAM(3)         0.0  FSTOP       |
| PARAM(4)       500.0  AUTOSTOP    |
| PARAM(5)         0.0  ORACLE      |
| PARAM(6)         0.0  FOCUS       |
| PARAM(7)         0.0  ANTS        |
| PARAM(8)         0.0  KERNEL      |
| PARAM(9)         0.0  CHARACTER   |
-------------------------------------

[     EVAL,    TIME]        OBJECTIVE FUNCTION VALUE         VIOLATION OF G(X)
------------------------------------------------------------------------------
[        1,       0]        F(X):         0.31862880         VIO:     0.000000
[      500,      34]        F(X):         0.31673280         VIO:     0.000000
[     1000,      68]        F(X):         0.31673280         VIO:     0.000000
[     1500,     102]        F(X):         0.31673280         VIO:     0.000000
[     2000,     136]        F(X):         0.31673280         VIO:     0.000000
[     2500,     171]        F(X):         0.31673280         VIO:     0.000000
[     3000,     206]        F(X):         0.31673280         VIO:     0.000000
[     3500,     240]        F(X):         0.31673280         VIO:     0.000000
[     4000,     275]        F(X):         0.31673280         VIO:     0.000000

OPTIMIZATION FINISHED  --->  MAXEVAL REACHED


        BEST SOLUTION FOUND BY MIDACO
--------------------------------------------
EVAL:      4000,  TIME:     275,  IFLAG:   1
--------------------------------------------
F(X) =                    0.316732800000000
--------------------------------------------         BOUNDS-PROFIL
x(   1) =                6.000000000000000;  %  ____________x_________
x(   2) =                3.000000000000000;  %  ______x_______________
x(   3) =                8.000000000000000;  %  ________________x_____
```

```
 MIDACO 4.0    (www.midaco-solver.com)
 -------------------------------------

 LICENSE-KEY:  MIDACO_LIMITED_VERSION___[CREATIVE_COMMONS_BY-NC-ND_LICENSE]

 -------------------------------------
 | N      3    | MAXEVAL        4000 |
 | NI     3    | MAXTIME       86400 |
 | M      0    | PRINTEVAL       500 |
 | ME     0    | SAVE2FILE         1 |
 -------------------------------------
 | PARAM(1)  0.001000  ACCURACY G(X) |
 -------------------------------------
 | PARAM(2)     43534.0  RANDOM-SEED |
 | PARAM(3)         0.0  FSTOP       |
 | PARAM(4)       500.0  AUTOSTOP    |
 | PARAM(5)         0.0  ORACLE      |
 | PARAM(6)         0.0  FOCUS       |
 | PARAM(7)         0.0  ANTS        |
 | PARAM(8)         0.0  KERNEL      |
 | PARAM(9)         0.0  CHARACTER   |
 -------------------------------------

 [     EVAL,    TIME]        OBJECTIVE FUNCTION VALUE         VIOLATION OF G(X)
 ------------------------------------------------------------------------------
 [        1,       0]        F(X):         0.31705632         VIO:     0.000000
 [      500,     294]        F(X):         0.30562032         VIO:     0.000000
 [     1000,     589]        F(X):         0.30562032         VIO:     0.000000
 [     1500,     884]        F(X):         0.30562032         VIO:     0.000000
 [     2000,    1179]        F(X):         0.30562032         VIO:     0.000000
 [     2500,    1474]        F(X):         0.30562032         VIO:     0.000000
 [     3000,    1769]        F(X):         0.30562032         VIO:     0.000000
 [     3500,    2064]        F(X):         0.30562032         VIO:     0.000000
 [     4000,    2359]        F(X):         0.30562032         VIO:     0.000000

 OPTIMIZATION FINISHED  --->  MAXEVAL REACHED


         BEST SOLUTION FOUND BY MIDACO
 --------------------------------------------
 EVAL:      4000,  TIME:    2359,  IFLAG:   1
 --------------------------------------------
 F(X) =                     0.305620320000001
 --------------------------------------------        BOUNDS-PROFIL
 x(   1) =                 6.000000000000000;  %  ____________x________
 x(   2) =                 3.000000000000000;  %  ______x______________
 x(   3) =                 8.000000000000000;  %  ________________x____
```