Shift, Store, or Sell:

Analyzing Scheduling Freedom in Smart Homes

A Green Computing Project Final Report

By

YE XU, WEI LI, ZHEFAN LIN, XI ZHANG

**ABSTRACT**

In this project, a “green” home house, partially or completely powered by renewable energy, would be managed according to difference types of workload and energy sources. An intelligent management system solver for home use is designed and implemented. Dataset from real houses powered by solar and wind energy is used. Simulation results of shifting the workloads, storing and selling the energy sources are provided and analyzed. Limitations and future works are discussed.

**Keywords**

renewable energy, scheduling, deferrable workload, shift, store, sell

# INTRODUCTION

The cost of electricity is increasing. In the past 25 years, the price of residential electricity has increased about 30%. [1] With more appliance used in resident home, the demand of electricity has increased almost 50% over 20 years. [3] According to the US Department of Energy (DOE), buildings accounted for about 38.9% of US primary energy consumption in 2006, 74% of which is electrical energy [4]. This electrical usage is roughly divided equally between residential and commercial buildings. Consequently, several efforts by the DOE and the research community [6] have begun to analyze energy use within buildings to identify the dominant energy loads. Recent research shows that depending on the special use modality of the building, the dominant electricity consumers can be lighting, computing infrastructure, or what is most often the case, heating ventilation and air-conditioning systems collectively referred to as HVAC [5, 6].

The discussion above is for governments’, organizations’, or companies’ concern. As a home consumer, however, the most direct concern might be the new market based price schemes compared to the traditional fixed price schemes [2]. Some states are using a peak/unpeak power demanding price scheme, costing more during peak time every day. Some others are using a real time hourly based price scheme, with different prices each hour.

This leaves the home consumer both advantages and disadvantages. They can reduce their monthly electricity cost due to this new market scheme. If properly used, the cost reduction can be significant. On the other hand, however, this work requires careful monitoring the real time price changes, as well as the detailed hourly appliance usage planning in advance, usually one day before. This clearly introduces extra burden on consumers. Additionally, consumers usually do not know the power usage patterns of appliances.

Of course the simplest way to cut electricity bill without such a burden is to use less electricity. But this way has negative impact on the quality of daily life. With the recently rapid development of embedded system area, technically every item in the physical world can be connected and controlled by micro-processors, sensor networks, and actuators. This sub area of embedded system is called Cyber Physical Systems, and becomes a hot research topic in both academic and industrial world. In addition, recently the space and installation cost for certain renewable energy source such as solar or wind is decreasing over the years. For example, the solar PV efficiency would represent a 50+% increment in space requirements. And the capital cost of solar energy has become substantially cheaper over time [1]. With these two rapidly developed technologies, it is possible to get the burden away from home consumers, which forms our main idea of this project.

To address the problems mentioned above, we implemented in this project an intelligent energy source and workload management system, that determines when to turn on or turn off appliance to achieve lowest electricity cost without disrupting daily life (Shift), when and how much to store low-cost energy for use during high-cost periods and when to use solar power or grid power (Store), and finally, when to sell additional energy back to net metering to get more benefit (Sell). This Shift, Store, and Sell management is the core part of this project. By analyzing the scheduling freedom in smart home, we are able to give suggestions to consumers the best possible scheduling plan, under different energy source choices.

The rest of the report is outlined as follows: section 2 gives some related work; section 3 discusses the theoretical part of our algorithm; section 4 demonstrates some simulations results and the relevant analysis; section 5 lists a couple of limitations of our project and some future work; section 6 concludes everything.

# RELATED WORK

The way of reducing electricity cost has been well studied in the research literature. In [1], Goiri *et al*, developed Parasol and GreenSwitch, based on the real datacenter experimental data. The authors discussed the tradeoffs of building a solar and/or wind powered datacenter in the future. Specifically, it discussed the space requirements and the capital cost of these technologies. This idea also can apply in resident home. They also demonstrated how Parasol and GreenSwitch works. For example, the authors introduced the infrastructure, hardware, software components of Parasol, with quite a few details. They also presented GreenSwitch, the core part of the system to manage workload and energy source. The authors explained in how to mathematically model each part of the system, as well as the objectives. Finally, the paper gave a couple of experimental results and evaluations.

The limitation of [1] when it is applied to home power usage is nonetheless obvious. Although it is a “kitchen-sink” approach, it only can be applied in data centers. Home appliances have completely different power usage patterns. For example, Data centers don’t have non pre-emptible loads, and all deferrable loads have the same patterns (period, deadline, etc.). These are the important assumptions made by the authors of [1], but they are clearly not true for home appliances, and surely cannot be ignored. Take dish washer as an example, once it gets started, it cannot be turned off until it finishes its job, which costs about two hours. This is one of the many typical non pre-emptible workloads. And another non pre-emptible load, wash machine, has completely different power patterns in terms of period, deadline, and power usage per period, etc.

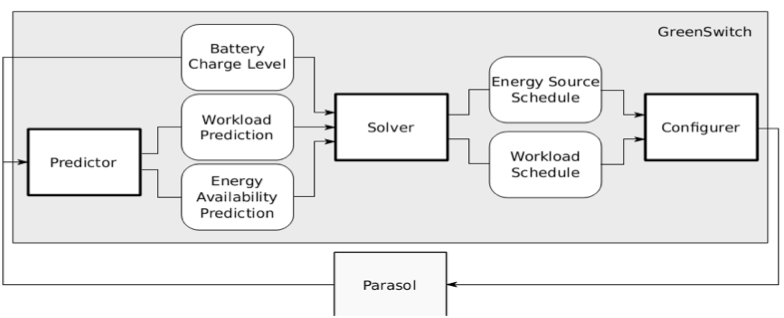
In [2], a SmartCharge scheduling algorithms was introduced. It is an intelligent charging and discharging system that determines when and how much to store low-cost energy for use during high-cost periods based on expectations of future demand. It analyzed, based on the data about peak/unpeak powers and different charge prices from different areas, the problems of traditional fixed rate charge plan for electricity, as well as the new problem for consumers based on the new market based charge plan. It presented SmartCharge, the core part of their intelligent charging system to manage battery charge and discharge schedule. The authors explained in detail how SmartCharge works, and how to mathematically model each part of the system, as well as the objectives. Then the paper gave a couple of experimental results and evaluations. In particular, the authors used real data from some houses for couple of months to show the simulation results and the corresponding cost reductions. Finally, the paper gave a cost and benefit analysis. In particular, the authors analyzed the return on investment (ROI) of the smart charge system, based on the installation fee of the hardware required for smart charge system.

Although the paper discussed in detail how smart Charge works, the paper ignores one important fact, that is, these days renewable energy source like solar and wind are more and more popular. Rather than buying battery array for storage, most of the home owners are more willing to invest their money on solar panels, and/or with battery array storage. Some home consumers only want to purchase solar panel only. So, rather than just considering about the battery array, which is unlike to be purchased alone, the author could have improved the practicality by offering the system with three options regarding energy source schedule: battery alone, solar only, and battery with solar. Another issue is that a lot of workloads in a typical home can be defined as “deferrable workloads”, such as refrigerator, AC, wash machine, dryer, etc. As long as they have enough power in each working period, the total work can be done. So regarding the workload scheduling, the authors could also have improved the whole cost reduction by adding this deferrable workload scheduling, which is also very common in today’s home appliance usage.

# DESIGN AND IMPLEMENTATION

As we discussed before, the limitations of [1] and [2] also gave us inspirations of the main idea of this project: extend the work of [1] and [2] for the smart homes, and analyze each scheduling combination in order to give home consumers best possible suggestions under different choices. The main parts of our scheduling algorithm include: Shift, Store and Sell. The main technique used in our project is Mixed Integer Linear Programming (MILP). In this section, we first discuss the architecture of the whole system, then we will give an overview of each part, followed by the detailed linear programming modeling of our algorithm, after which some of the tricky techniques are discussed for the modeling.

## Architecture and Overview of the System



**Figure 1: Parasol Architecture [1]**

Above figure indicates the architecture of parasol [1]. Since this architecture is made highly reusable, we include it here for the big picture of our specific solver design. Below is the brief explanation of some of the main parts of this architecture.

**Predictor:** This component predicts the workloads and the renewable energy production.

**Solver:** This component takes the prediction and the current battery charge level from predictor as an input, and outputs a workload schedule and an energy source schedule. In our project, we redesigned the solver part in order to support home appliance usage.

**Configurer:** It effects the changes prescribed by the solver, and it is the only part specific to the system GreenSwitch is supposed to control.

**Overview of our Solver:** The solver part of the architecture is our main concern. Basically, it can handle grid electricity cost, on/off peak different charges. Also, there are two types of workloads in a typical home, non-deferrable and deferrable workloads. For the deferrable loads, it can be further divided into pre-emptible and non-pre-emptible loads. The examples of non-deferrable loads are TV, Computers, Stereos, Play Stations, etc. The examples of pre-emptible loads include AC, Refrigerator, etc. And the examples of non-pre-emptible loads include Dish Washer, Washing Machine, Dryer, etc. Finally, there are three types of energy sources we can schedule, solar, battery array, and grid power.

**Objective of our Solver:** Under non-deferrable workloads we should only schedule the use of the energy sources to minimize the grid electricity cost. For deferrable workloads we are able schedule both workload and use of the energy sources to minimize the grid electricity cost. The workload schedule determines how much energy should be consumed in each cycle. And the energy source schedule plans how much energy to draw from each source in each cycle.

## Models, Optimization, and Solution Approach

In this part, we will formulate the optimization problems for the solver. We provide a detailed linear programming modeling here, separated by Shift, Store, and Sell part of our algorithm.

Before we start our linear programming modeling, it is necessary to mention here all the variables and their corresponding meanings are listed below.

**Parameters out of our control:**

* Load: average predicted required power for each time interval, in kWh. (If the workload is deferable, we might add a new variable WorkLoad as the offered load in each time interval)
* T: number of time intervals
* BattCapa: battery's usable capacity, in kWh
* BattE: battery charging efficiency
* GridCost: grid energy price in real time, in cents per kWh
* Green: amount of predicted green power available in each time interval
* α: percentage of retail price paid in net metering

**Variables under our control for optimization:**

* LoadGreen: amount of green power to be used for load
* LoadGrid: amount of grid power to be used for load
* LoadBatt: amount of battery power to be used for load
* BattGreen: amount of green power to be used for charging battery
* BattGrid: amount of grid power to be used for charging battery
* NetGreen: amount of green power to be used in net metering
* Grid: amount of grid power to be used for any purpose
* bin: ensure mutual exclusive
* preemptibleLoadsSchedule
* nonPreemptibleLoadsSchedule

### **Store and Sell: Modeling Energy Sources**

We start by modeling the Store and Sell part of our algorithm. Basically, store means how much energy source to store or to use for each time period, whereas sell means when to sell additional energy back to net metering to get more benefits. That being said, the store part is essentially modeling the energy sources.

There are three sources to power the house which are Battery, solar energy and grid.

**Modeling Battery:**

For modeling battery, we first assume total battery charge rate cannot be higher than BattCapa/4,

Second, the power discharged from the battery is never greater than the power charged to the battery:

Third, the energy stored in battery, which is the difference between the energy charged to or discharged from the battery over the previous time intervals, cannot be greater than its capacity:

Fourth, we cannot use the batteries and do net metering at the same time:

Fifth, we cannot charge and discharge the battery at the same time:

**Modeling Solar and Grid Power Source:**

The renewable power solar energy may be used to run the workloads, to charge the battery, and/or to sell back to net metering:

The grid can be used to power the load and charge the battery:

There is one more constraint regarding the grid power and solar energy source scheduling, i.e. we cannot draw from the grid to power the load at the same time as doing net metering:

### **Shift: Modeling Workloads**

The modeling of workloads is divided into two parts: non-deferrable workloads and deferrable loads. But first of all, all of the workloads for each time period should be powered by at least one of the three energy sources, which gives us the first equations of workload modeling:

**Non-deferrable Loads:** In (9), the non-deferrable workload is predicted by predictor in Figure 1 in advance. But since in our project we didn’t design the predictor, we made an assumption that we know next 24 hours’ hourly power usage pattern, and hard coded into our program as the input to our solver.

**Deferrable Loads**: As for the deferrable loads, since every appliance has different power usage patterns, we need some data structure to distinguish such patterns. The most interesting characteristics describing the power usage patterns are the appliance’s period, relative deadline, running time, and power usage per period. Thus, we define a tuple called pattern tuple as: [period, deadline, running time, power per period] to represent each appliance.

**Pre-emptible Loads:** Now we will discuss about the pre-emptible work load scheduling. The appliances such as refrigerator and AC have their own periods. During each period, we could shift their load as long as the total power usage per period is greater than or equal to the required power usage per period. This gives us the following equation for pre-emptible load scheduling:

Equation (10) gives us one appliance’s scheduling; we need to add up all of the pre-emptible loads at each t, which gives us the total pre-emptible load at each t.

Note that for the simplicity, this project didn’t take into account the relative start time for each load at each period. For example, AC’s period maybe 12 hours and running time 5 hours, which means AC needs to be turned on for 5 hours from 0 am to 11 am, as well as from 12pm to 11pm. But typically, AC might not be turned on at the starting point of each period, a.k.a, at 0am or 12pm, it might start working some time later than the starting point of each period, which needs a relative start time. But for simplicity we assume AC can be turned at the starting point of each period.

**Non-pre-emptible Loads:** Since non-pre-emptible loads cannot be stopped once it starts, we can only control the start time of each load. The formal scheduling for non-pre-emptible loads is the following equation:

(11)

### **Optimization Goals**

Our model goal is achieving minimum electricity goal, the objective function is as follows:

## Dealing with Mutual Exclusive Problems and Non-pre-emptible Loads

**Mutual Exclusive Problems:**

As we would notice, some constraints are not linear, for example, the following three are called mutual exclusive constraints, which means two conditions cannot happen at the same time:

* We cannot use the batteries and do net metering at the same time
* We cannot draw from the grid to power the load at the same time as doing net metering
* We cannot charge and discharge the battery at the same time

Now we show how to solve this problem. The trick is to add one more variable called bin, a binary number either 0 or 1. Now if we want to model the following problem: given x > 0, y = 0 or vice versa, we can convert it to the following five linear constraints:

If bin is 0, the above three equations can be interpreted as:.

If bin is 1, the above three equations can be interpreted as:.

It is now clear the original constraints have been converted to linear constraints using this trick.

**Non-pre-emptible Loads:**

As we can see constraint (11) is also nonlinear, and it is difficult to be converted to linear model using any tricks. In order to deal with this situation, we could instead perform the following approach: we don't need the variable startTime. If the period of the load is 24 hours, I need (24 – runningTime) objective functions. For example, the first objective function would be: min (Cost of all the workloads we have done so far + Cost of nonpreemptibleLoads from time 0 to runningTime). This objective function means we just need to assume (not schedule) Load\_1 = Load\_2 = Load\_3 = ... = Load\_runningTime = PowerPerPeriod / runningTime, other Load\_i are 0. In other words, we convert it to a non deferable load starting at time 0 and up for runningTime. We generate (24 - runningTime) such objective functions, and get an optimized total cost for each one, then we select the min cost out of these objective functions as our final solution.

This approach has a substantial running time if the number of non-pre-emptible loads is big, or if the number of periods for every day is more than 1(period is less than 24). But we can make an assumption that typical appliances like dish washer, wash machine, dryer, they normally only need to be run at most once a day, and there are not a lot such kind of non-pre-emptible appliances in a typical house.

After modeling the constraints and objective functions, we used an open source Mixed Integer Linear Programming solver called OPTI MILP solver [8]. The reason for using this solver is that the additional variable bin should be either 0 or 1, and it should be integer.

# Experimental Evaluation

## Methodology

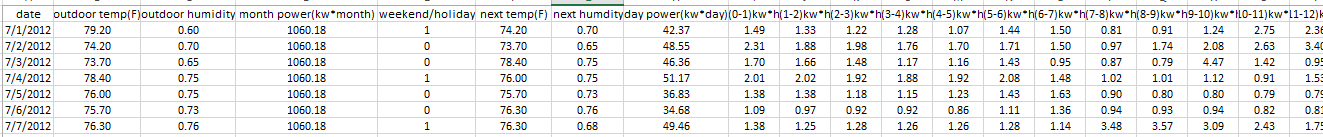
In order to quantify the potential for saving based on the electricity market pricing plans, we need data about the home house electricity consumption. Additionally, the environmental data provided by the weather sensors can be used to predict the aggregate consumption for homes.

Researchers from UMASS Amherst computer science department made publicly available numerous data sets in [3]. The data sets include three homes’ electricity consuming value during the period between April and July in 2012.

Take the data observed from home A for example. Home A is located around Amherst. The data sets collected the energy consumption of home A for 68 days. For each day, dataset came from 26 indexed different sources, including grid, HRV, washing machine, bedroom lights and so on. We filtered only the average real and apparent powers every second from grid supply which represents the aggregate consumption per second for the whole home house. Every day has 86 thousands and 4 hundreds seconds but the amount of the filtered data is less than this amount. In addition, the timestamp for each data is unix time which means that how many seconds have passed since January 1st 1970 however we found that the timestamps are not correct. So we used MATLAB to process the filtered data by inserting data into those lost seconds as same as the last reasonable value and shifting all data to correct starting place for every day according to the timestamps. Finally, we calculated interval power (power per hour), day power and month power.

And then we collected the outdoor temperature and outdoor humidity for every day from the website www.wunderground .com which contains the history weather data collected from the sensor that is located in UMASS Amherst computer science building. Also we labeled weekday as “0” and weekend or holiday as “1” that can be convenient for future process as data input.

The following figure is part of the final results after the processes mentioned above:



**Figure 2: Sample Processed Data**

## Experimental Results

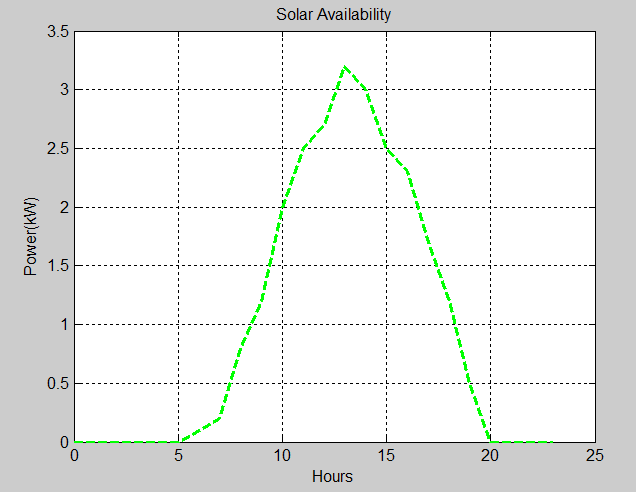
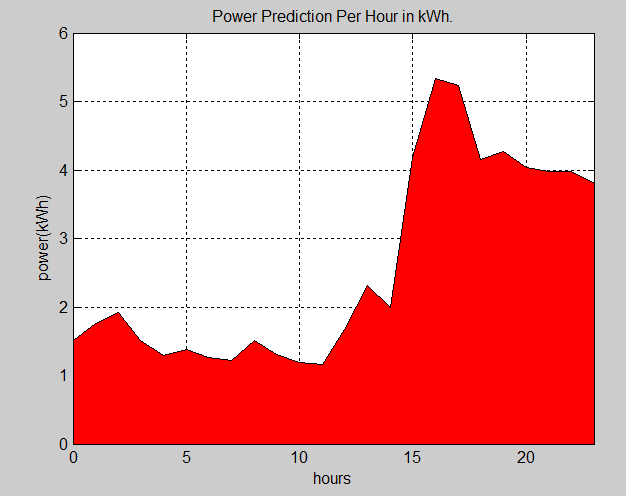
We need to do some assumptions at first. First, the battery is not empty at the beginning of every day. Second, we assume that our system can behave like an oracle, in other words, it knows next day’s power pattern and solar availability during that day. Third, the period is integer and it can be divided by the execution time T. Forth, the deadline is equal to period.

These are the data fundation for the testing and simulation. We choose one day’s data form the data sets. The first one is the power pattern per hour for the whole day. The second one is the solar prediction for that day. In common sence, solar strength during the 12pm and 2 pm is the highest. The third figure plots the electricity grid price per hour for each day according to the electricity market pricing plans which are fixed. Additionally, we choose three appliances in house for deferrable modeling, including Central Air Conditioner, refrigerator and dishwasher. Among them, dishwasher is not pre-emptible. The parameter for each appliance is as following:

ACCentral = [24,24,8,56]

Refrigerator = [2,2,1,0.36]

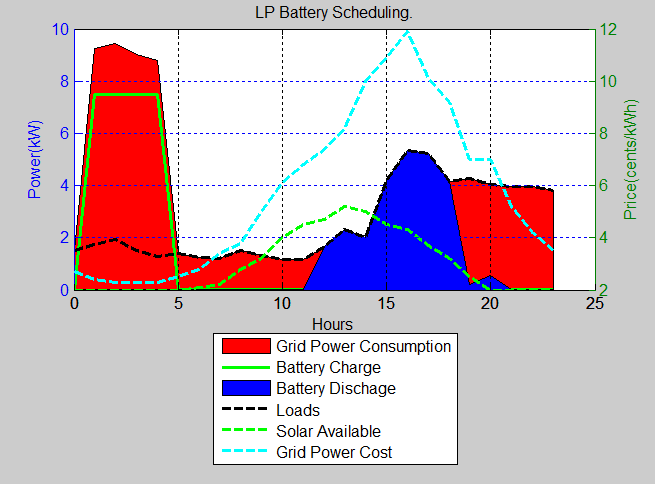
Dishwasher = [24,24,2,4]

Note [deadline, period, execution time, power per cycle]

**Figure 3: Power pattern per hour, Total Non-Deferrable Available Solar per hour, and Grid Price per hour**

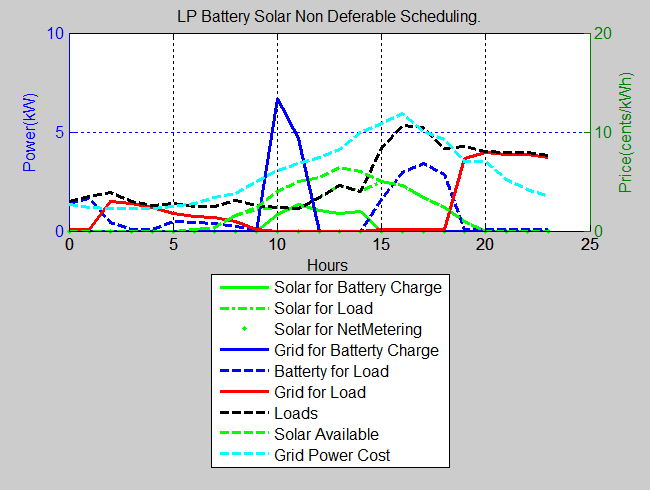
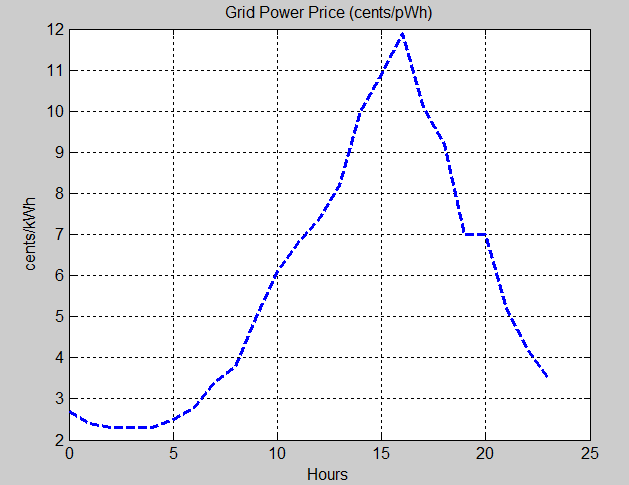
**Battery Only Scheculing Simulation:**

This is the simulation result for the battery only scheduling which means that we just use battery and grid supply to power the whole home house. The area below the black line is the power demand which has been plotted in the data fundation. The red area is the grid power consumption. From this figure, we can see that, during 0 am and 5am which is the low-cost period based on the grid power pricing plan, we use some grid power to supply for the power demand, the extra grid power is used to charge the battery which will be in turn used during the high-cost period that is blue area in this figure. Under this scheduling, the electricity bill is $2.3 with the battery only, compared with the $4.17 which is the original electricity bill without using the battery. The total cost reduction is up to 44%.



**Figure 4: Battery Only Scheculing Simulation**

**Battery Solar Scheduling Simulation:**

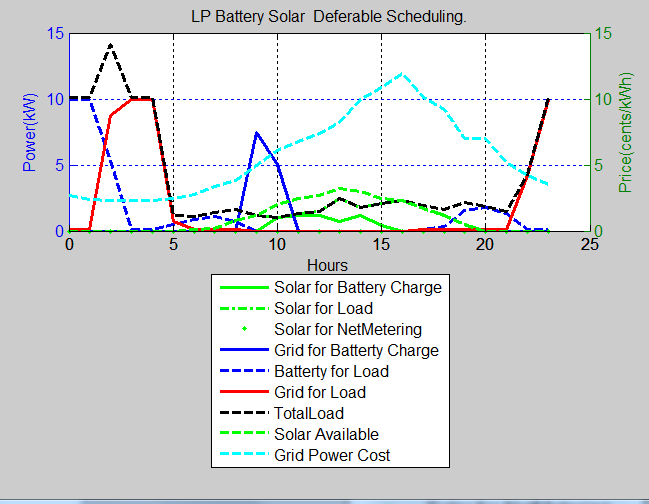


**Figure 5: Battery Solar Scheduling Simulation:**

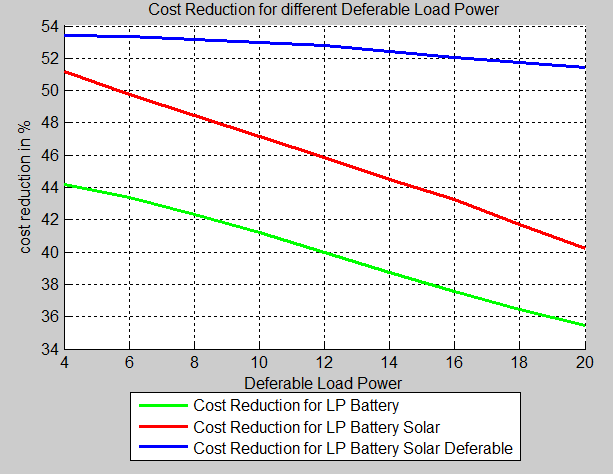
In this simulation, we add the solar to the previous power supply group which includes the grid and battery. Compared with the last figure, because we know that in that day, we can use solar availability, we do not need to charge the battery a lot during the night. Instead we use solar to charge the battary during the day when the solar is available. When the peak power demand is coming, the solar and battery both supply electricity power. The resulting electricity bill under this scheduling is $1.98 which is reduced by 52%.

**Battery Solar Plus Workload Scheduling Simulation:**

Other than battery and solar for supplying, we use the deferrable modeling to schedule the workload as well. From the diagram, the black line which respresents the workload is changed by shifting the deferrable workload to the low-cost period so that getting the bill reduction. This scheduling can save the most which is 54%.



**Figure 6: Battery Solar Plus Workload Scheduling Simulation**



**Figure 7: Cost Reduction for Different Deferrable Load Powers**

**Cost Reduction for Different Deferrable Load Powers:**

Figure 7 shows the cost reduction when we use different method on system to save electricity cost. We can see that, without adding deferrable function in our system, cost reduction decreases when more load power in system. After adding deferrable function, we can achieve more than 50% at 50kWh load power. This graph also illustrates the strategies we can use for systems with different load power. For example, if a house only consumes 4 kWh, we can only use Battery and Solar energy without deferrable function to achieve at least 50% electricity cost reduction. It means more saving on construction stage.

# LIMITATIONS AND FUTURE WORK

This system still has some limitations. First, we did not complete the design and implementation of predictor and configure. If predictor works based on previous data, our simulation will be more accurate. In our current implementation we just assumed the solver knows in advance all of the data needed. Predictor design can be achieved using some Machine Learning techniques. The second limitation is that the cost reduction is for only one day. The average cost reduction was not simulated. But it can be achieved by doing multiple simulations. Another limitation is trade off analysis. Based on different houses with different powers, it is better to know the hardware cost of different cost reduction strategies. So we can use the right strategies to achieve maximum reductions in cost of building and electricity. For example, different batteries have different capacities and costs, usually the cost increases with capacity. Therefore, using suitable battery can minimize building cost and increase utilization of battery capacity.

# CONCLUSION

In this project, we designed a MILP solver to schedule both energy sources and workloads, both deferrable and non-deferrable loads, both pre-emptible and non-pre-emptible loads. For each Deferrable Load, it has its own pattern. We compared different cost reduction benefits with different scheduling algorithms. Simulation results also showed when to use which scheduling can achieve more benefits.

# REFERENCES

1. Íñigo Goiri, William Katsak, Kien Le, Thu D. Nguyen, and Ricardo Bianchini. [*Parasol and GreenSwitch: Managing Datacenters Powered by Renewable Energy*](http://www.cs.rutgers.edu/~ricardob/papers/asplos13.pdf)*.* [ASPLOS 2013](http://asplos13.rice.edu/)
2. Aditya Mishra, David Irwin, Prashant Shenoy, and Jim Kurose, *SmartCharge: Cutting the Electricity Bill in Smart Homes with Energy Storage*, e-Energy 2012, May 9-11 2012, Madrid, Spain.
3. Sean Barker, Aditya Mishra, David Irwin, Emmanuel Cecchet, Prashant Shenoy, and Jeannie Albrecht. *Smart\*: An Open Data Set and Tools for Enabling Research in Sustainable Homes.* Proceedings of the 2012 Workshop on Data Mining Applications in Sustainability (SustKDD 2012), Beijing, China, August 2012.
4. *DOE. Buildings Energy Data Book*, Department of Energy, March 2009. <http://buildingsdatabook.eren.doe.gov>
5. Y. Agarwal, B. Balaji, R. Gupta, J. Lyles, M. Wei, and T. Weng. *Occupancy-Driven Energy Management for Smart Building Automation*. In Second ACM Workshop on Embedded Sensing Systems For Energy-Efficiency In Buildings, 2010
6. Y. Agarwal, T. Weng, and R. Gupta. *The Energy Dashboard: Improving the Visibility of Energy Consumption at a Campus-Wide Scale*. In First ACM Workshop on Embedded Sensing Systems For Energy-Efficiency In Buildings, 2009
7. Stephen McLaughlin, *Patrick McDaniel, William Aiello, Protecting Consumer Privacy from Electric Load Monitoring*, CCS’11, October 17–21, 2011, Chicago, Illinois, USA
8. OPTI MILP solver, <http://www.i2c2.aut.ac.nz/Wiki/OPTI/index.php/Probs/MILP>