GreenSwitch: The Home Extension

A Green Computing Project Proposal Presented

by

YE XU

ZHEFAN LIN

WEI LI

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 extension of “GreenSwitch” for home use is designed and implemented. Dataset from real houses powered by solar and wind energy is used. The initial approach and a rough timeline for the entire project are proposed in this document.

**Keywords**

Renewable energy, scheduling, home house, GreenSwitch

# INTRODUCTION

Several companies have recently announced plans to build “green” datacenters, i.e. datacenters partially or completely powered by renewable energy. For example, Apple is building a 20MW solar array for its North Carolina datacenter [6]. McGraw-Hill has recently completed a 14MW solar array for its datacenter [7]. Our idea is that why don’t we use this approach to build a “green” home house. These home houses will either generate their own renewable energy like using solar energy or draw it directly from an existing nearby plant. However, solar energy is intermittent, which requires approaches for tackling the energy supply variability. One approach is to use batteries and/or the electrical grid as a backup for the renewable energy. It may also be possible to adapt the workload to match the renewable energy supply. For highest benefits, green home house operators must intelligently manage their workloads and the sources of energy at their disposal.

# CURRENT PROBLEM AND RESEARCH

Given the emissions and increasing societal awareness of climate change, governmental agencies, non-profits, and the public at large are starting to demand cleaner products and services. According to the US Department of Energy, buildings accounted for about 38.9% of US primary energy consumption in 2006, 74% of which is electrical energy [3]. This electrical usage is roughly divided equally between residential and commercial buildings. Consequently, several efforts by the DOE and the research community [5] 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 [4, 5]. In [1] Goiri *et al.* designed a “GreenSwitch” system to control their real micro data center “parasol”, based on linear optimization algorithms, which was claimed can highly offset the electricity cost as well as installation cost. We argue in this project that a “green” home rather than a data center could also be benefitted by an extension of such a management system.

# INITIAL APPROACH

## GreenSwitch: the Home Extension

In [1], Goiri *et al.* demonstrated the central real time control system, GreenSwitch, as their core system to make benefit out of renewable energy plant investment with solar and/or wind power. Rather than controlling the massive data center, part of the contribution in this project is to adapt GreenSwitch so that it will be able to manage workload and energy source in homes/buildings, which leads to GreenSwitch: the Home Extension.

The solver of original GreenSwitch was mainly based on linear programming (LP). Linear programming is a well-developed area, and the algorithms are available to solve various problems. Because of this general technique, as well as the whole GreenSwitch architecture, future researchers are able to reuse this system without significant modification; only the configuring part and the mathematical modeling of specific optimization problems are needed to be changed.

We now discuss about the modeling of LP in home appliances. In [2], Barker *et al.* showed that a typical home has the following appliances: a central air conditioning (A/C) or separate window A/C units or HVAC unit, an electric dryer and washing machine, heat recovery ventilation (HRV) unit, dishwasher, refrigerator, and freezer. In addition, the building also has a solar panel, generating power and feeding into the home’s grid supply. To model LP, we need to model both objectives and constraints. Note that in [1], the solver part of GreenSwitch is essentially the design and implementation of LP. As for the predictor, the technique would be similar; thus we will skip for the proposal.

The objectives of the solver should also be similar to the original GreenSwitch; the solver should provide an optimized workload schedule and energy source schedule, under which it should minimize the total electricity cost in a certain range of time.

The deferrable and non-deferrable workload determination, however, is an interesting problem. Since traditional non-deferrable workload refers to time critical system tasks such as hard real time systems, very few of these home appliances has to be treated like so. However, some systems, such as HVAC, does need to be turned on and off once the sensor senses the values outside certain defined range. Yet even if such system fails to start or finish its job before its deadline, there is no severe harm happening. In fact, this boundary between deferrable and non-deferrable is highly vague and also depended on particular custom needs. Thus in our system we give customers the power to choose which one can be deferred and the relevant deadlines.

As for the constraints modeling, it is system specific with an entirely different set of parameters and equations. But based on certain common senses, such as total offered energy should be equal to or greater than the total requested energy, it should be straightforward to implement. The configuring part of GreenSwitch is rather a bunch of actuators which adjust different appliances according to the commands from the solver. The setting up highly depends on the physical limitations and saturation areas of these actuators. For now, we just assume that every appliance under control can be delayed indefinitely, every energy source can be activated/deactivated and the maximum amount can also be changed.

## Dataset

Researchers from University of Massachusetts Amherst Computer Science department made publicly available numerous data sets in [2] for enabling research in sustainable homes. The dataset came from a variety of different sources, including, electricity usage at the mains panel, each circuit, and plug load. Additionally, the data also came from multiple weather, motion, door, wall switch, and thermostat sensors, as well as electricity generation data from solar panels and wind turbines. We observe 6 types of data as follows:

* Electricity at the Mains Panel: average real and apparent power every second for the home and each circuit at the mains panel.
* Electricity at Outlets: real power usage at intervals from home’s plug loads.
* Wall Switch Events: on-off-dim events at wall switches.
* Average electricity generation from solar panels and micro wind turbines every five seconds.
* Thermostat Events, Motion Events and Door Events: a variety of events relating to energy consumption, including motion sensors, door/trigger sensors, and thermostat sensors.
* Weather Station Data: environmental data provided by the weather sensors every minute both inside and outside the home.

With the data provided by the team, the following work can be done:

**Cost Optimization**: 1st, Use the weather data to predict the aggregate consumption for homes; 2nd, quantify the potential for savings based on the today’s electricity market pricing plans.

**Demand Flattening**: Use a Least Slack First (LSF) to schedule loads in ascending order of their remaining time without affecting their objective. In addition, use home electricity data, plug load and circuit data for eight background loads, and the team’s temperature and humidity data from the weather sensors to evaluate LSF’s potential for demand flattening.

**Load Monitoring**: Use AutoMeter to resolve a home’s electricity usage into several parts each second with low resolution data, from Insteon wall switch events and iMeter plug loads.

**Renewable Prediction**: Predict future renewable generation using weather forecasts from the National Weather Service.

**NILM**: Although Non-intrusive load monitoring (NILM) focuses on large scale scenarios, commonly greater than 100 loads, there are still many relatively low-power loads, like less than 50 W. Low power loads is a common characteristic of homes. By the analysis of the team’s data, we find it useful in developing and evaluating new disaggregation algorithms for electricity data.

## Simulator

Even though we can get the power consumption and energy supply data from some sustainable homes, we cannot control the on/off status of those appliances used in these home and those backup energy supply devices, such as batteries and solar panel. Thus we need a simulator so that we can directly simulate and compare the results of various conditions.

Functions of the simulator are as follows:

* Import the input data
* Set the condition parameters, such as the guard band of energy consumption. We can also assume that we have batteries, solar panel or wind panel.
* Make scheduling and management of the home workloads and energy source according to GreenSwitch: the Home Extension.
* Show the energy use of the whole day and account the total costs. Also show the cost reduction using our GreenSwitch: the Home Extension system.

# ROUGH TIMELINE

We would like to employ an agile development cycle to keep positive progress continuously, since this application system is essentially for home customer’s use, if commercialized. Each week we plan to make progress with feedback from our customers and mentors; in this case we hope our professor for this project could serve as the customer’s role. Roughly speaking, we have in mind the following timelines:

1. Some of our group members are going to spend two or three weeks doing research on linear optimization problems, and model the objectives and constraints. (doesn’t need to be accurate since we are doing agile). Note that this part is the controlled plant as well as the solver component of the system.
2. At the same time, the rest of our members are looking into the sensor and actuator markets, figuring out physical limitations and saturations of sensors and actuators, cooperating with our algorithm and modeling team to design a realistic system. Note that this part really is the configurer component of our system.
3. Make a quick and dirty implementation, get feedback, and then improve accordingly. Repeat this cycle until everybody is happy.

# 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. 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.
3. DOE. Buildings Energy Data Book, Department of Energy, March 2009. <http://buildingsdatabook.eren.doe.gov>
4. 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
5. 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
6. Data Center Knowledge. Apple Plans 20MW of Solar Power for iDataCenter, 2012. <http://www.datacenterknowledge.com/archives/2012/02/20/apple-plans-20mw-of-solar-power-for-idatacenter>
7. Data Center Knowledge. Data Centers Scale Up Their Solar Power,2012, http://www.datacenterknowledge.com/archives/2012/05/14/data-centers-scale-up-their-solar-power/