An Integrated Architecture for Demand Response Communications and Control

Michael LeMay, Rajesh Nelli, George Gross, and Carl A. Gunter
University of Illinois Urbana-Champaign
Email: {mdlemay2,rnelli2,gross,cgunter}@uiuc.edu

Abstract

In the competitive electricity structure, demand response programs enable customers to react dynamically to changes in electricity prices. The implementation of such programs may reduce energy costs and increase reliability. To fully harness such benefits, existing load controllers and appliances need around-the-clock price information. Advances in the development and deployment of Advanced Meter Infrastructures (AMIs), Building Automation Systems (BASs), and various dedicated embedded control systems provide the capability to effectively address this requirement. In this paper we introduce a Meter Gateway Architecture (MGA) *to serve as a foundation for integrated control of loads by energy aggregators, facility hubs, and intelligent appliances. We discuss the requirements that motivate the architecture, describe its design, and illustrate its application to a small system with an intelligent appliance and a legacy appliance using a prototype implementation of an intelligent hub for the MGA and ZigBee wireless communications.*

Index Terms

Demand Response, Advanced Meter Infrastructure, AMI, ZigBee, Building Automation System, BAS, Home Automation, Meter Data Management

1. Introduction

Fluctuations in electrical loads require generation investments to enable Electricity Service Providers (ESPs) to provide reliable service around-the-clock. Reducing the peaks in these fluctuations could produce significant costs savings and reduce risks of outages, thus improving efficiency and reliability of ESPs. In principle this can be done with *demand response* techniques in which electricity users take measures to reduce demand during times when generation is, or is likely to be, expensive. Unfortunately, these benefits of demand response have not yet been widely realized in practice. There are several reasons for this, but one of the most important is the lack of price-responsive controls over important loads. However, new technologies are quickly reaching a stage in which a much greater range of loads could support demand response. Three important examples of such new technologies are are Advanced Meter Infrastructure (AMI), Building Automation Systems (BASs), and embedded control systems.

AMI provides intelligent metering based on data network communications to facilities. The current driver for much of the AMI deployment is reduced billing costs for ESPs, but many AMIs can also support demand response by providing Real-Time Pricing (RTP) to load owners. BASs are being deployed in the next generation of "smart buildings", which typically provide a network system such as wireline Ethernet to control functions like security and HVAC. A closely related technology for building and home automation based on wireless communications using ZigBee has been reaching fruition as well. These technologies provide communication and control that could be used to support demand response by transmitting pricing information from the AMI to facility load controllers. Finally, advances in networked embedded control devices provide an avenue for appliances to carry out their own demand response actions based on wireless monitoring of electricity prices or react to controls from elsewhere.

Today's advanced BASs provide excellent capability to fine-tune the set points of energy utilization in various heating, cooling, and lighting loads in edifices. The EnergyPlus simulations at the New York Times Headquarters building [1], demonstrate the extension of BASs for performing direct load control. The simulation consisted of a demand response event lasting four hours and produced a maximum demand reduction of 400 kW in the NYT building. Further, a different direct load control option available to Energy Service Providers (ESPs) is to make use of AMI and remotely disconnect or cycle loads of customers who sign up for such programs. For example, the Energy$mart Thermostat^{SM} pilot program launched by SCE [2] makes use of a two-way programmable thermostat connected to the internet and controlled via radio to tune the set-point of air conditioners in the interval of [2,4] degrees Fahrenheit. The metered data collected from 4600 such thermostats during the tests conducted in the summer of 2003 showed an average energy savings of about 6.0 MWh during the first hour of the field trials and a peak demand reduction of 9 MW during the initial 15 minutes of the hour, for a setpoint increase of four degrees in all the thermostats. The Grid Friendly Appliance controller developed by Pacific Northwest National Laboratory (PNNL) is an example for the use of an embedded control system in direct load control. It senses the grid conditions by monitoring the frequency of the system and provides automatic demand response to meet system needs. It can decrease the residential load by turning household appliances off for a specified period - typically on the order of a few minutes, hence raising the grid frequency so as to return to 60 Hz operation [3].

Although these and other studies show promise for individual strategies for demand response, none describe or use an integrated approach to allow their deployment to effectively and economically implement demand response on a large scale. The aim of this paper is to propose and take initial steps to validate an architecture that integrates demand response based on AMI, BASs, and embedded control systems. The main technical challenge is to accommodate within the same framework the existence of multiple loci of control for demand response, such as smart appliances, BAS hubs, and centralized authorities like independent energy aggregators or the electricity supplier itself. We describe an approach in which these can be integrated into a single architecture. This architecture, which we call the "meter gateway architecture", is based on the deployment of the facility meter as the primary communication system linking the BAS to the AMI.

Our demonstration illustrates how incremental adoption of demand response can be achieved with existing infrastructure and embedded devices in such a way that the resultant system is capable of automatically responding to fluctuating price signals in an intelligent and flexible manner. To this end, we developed a collection of prototype demand response systems. They all respond to fluctuations in real-time prices downloaded from the web and metered using AMR, to emulate an AMI solution. The first prototype is a laptop that monitors energy prices and uses this information to regulate the times at which it recharges its battery. This explores demand response by intelligent embedded controls and the potential use of storage systems. A second prototype is a hub that uses a wireless link to enable demand response to a device that has none on its own. In particular, this system uses an RTP-enabled thermostat to override settings on an air conditioning unit using X10 home automation. This models both "smart thermostats," which are an area of current experimentation, and strategies for creating demand response in legacy appliances. We argue that the architecture provides a general strategy for incremental adoption of demand response that effectively exploits the emerging AMI, BAS, and embedded control technologies. We even provide preliminary evidence that such controls could lead to incremental cost savings for individual homeowners that have been difficult to achieve to date due to the past requirement that homeowners manually respond to price fluctuations.

The paper contains six additional sections. In Section 2 we motivate demand response and justify the idea that an architecture based on multiple loci of control is needed. In Section 3 we describe the meter gateway architecture that fulfils this requirement. In Section 4 we describe our prototype system and experiments aimed at validating this architecture. In Section 5 we discuss the implications of the prototype and experiments with respect to this and other potential architectures in a broad context. Section 6 summarizes related work and Section 7 concludes.

2. Motivation

Demand response is the concept that load controllers may vary their usage levels to accommodate changing electricity prices or to avert system instability. Fluctuations in electricity usage can be substantial in current high-load areas like cities and industrial parks, and the reserves required to meet peak demands introduce significant costs, causing the wholesale price of electricity to vary by a factor of two or more from one hour to the next, and a corresponding but dampened response in subsequent day-ahead price estimates. If load controllers are aware of this fluctuation, then there is an opportunity for significant savings from delaying demand or making incremental compromises in comfort or convenience. For instance, during a peak period it may make sense to allow thresholds for air conditioning (a large contributor to peak demands) to vary outside the usual range by a few degrees. If price information is made available to load controllers and the owners of these loads are allowed to reap the benefits of trimming demand during peak periods, then there is a strong free market incentive for effective demand response. In this section we explore aspects of the technological design space to enable this capability by describing key aspects of the infrastructure required to support it and the likely loci of control for demand response that could use this infrastructure.

AMI is an emerging technology growing from AMR. The main goal of AMR was to reduce the costs of reading electrical meters but AMI provides the promise of other capabilities based on bidirectional communications where data can be sent to a meter as well as retrieved from it, and, in some cases, the ability to execute control actions (such as shutting off electricity). AMI can be viewed as a communication and control link between a *Meter Data Management Agency (MDMA)* and a collection of metered facilities. Advantages of AMI include not only the cost savings of AMR, but also prospects for increased customer control, including demand response. AMI brings to distribution network end users an analog of the digital communications that Supervisory Control and Data Acquisition (SCADA) systems provide for substations in the distribution and transmission networks. AMR and AMI systems can use a variety of communication mediums. On one end of the spectrum, a simple type of AMR uses short range radio links that require readers to drive by facilities in vans to collect readings. At the other extreme, many sophisticated AMI solutions are now using mesh networks where neighboring meters communicate data for one another to route information to and from the MDMA, which is attached to one or more nodes on the mesh. Our discussion here assumes that the AMI is at least capable of sending real-time pricing information to meters and recording power usage in accordance with this pricing. This rules out the van-based AMR systems, but includes networks that use technologies like Power Line Carrier (PLC) communications or cellular links, as well as robust mesh networks.

BAS is an emerging technology for controlling networked computers, sensors (like motion detectors) and actuators (like door locks) in facilities. These technologies fall into two general groups. One is aimed at office and factory facilities, including sophisticated features like tracking movement of people and objects or controlling specialized machinery. The second is aimed at residential facilities, where key objectives include controls for appliances, such as light dimmers and emergency monitoring to detect break-ins or fires. A particular case of interest is "smart" thermostats, which provide a computer that controls the home HVAC system and are, in some cases, able to communicate with non-HVAC household appliances. A major trend in the home automation market is the use of short range wireless technologies. The mainstream home automation space is dominated by the X10 protocol, which was introduced in 1978 and uses in-home PLC to control plug-in modules using binary commands transmitted at 60bps. X10 has been extended to support wireless remote controls by using transceivers that convert RF control commands into X10 PLC signals. Several alternatives to X10 have been developed, but most retain backward-compatibility with the basic X10 protocol. However, X10 now faces new competition that could fundamentally change the communications model for home automation. ZigBee is a link layer based on the IEEE 802.15.4 MAC designed for this application space. It improves over technologies like WiFi (802.11) in power consumption and robustness, and may have a disruptive influence on office and factory BAS systems, which now are based mainly on wireline Ethernet, but may in the future use mesh networks of ZigBee nodes [4].

Another trend is the deployment of smart appliances that include some form of embedded control. This trend is not new, and, increasingly, appliances like washing machines and ovens have embedded control. A newer trend is the addition to these devices of a network communication link and possibly attachment to the Internet. An interesting illustration of this is the idea of a networked microwave oven that can read a bar code on a package and download cooking instructions from the Internet for the food in the package [5]. The relevance of this type of communication is its potential to allow the appliance to collect information from outside its own sensor system and exploit this for various purposes, such as energy savings through demand response. Consider, for example, the idea of a dish washer that includes in its energy-save mode a strategy for doing the wash when the electricity price is below average (for example, it might learn how to predict the best moment for washing in the 3 hour interval following a wash request).

Appliances that have rechargeable batteries provide a special opportunity since they can, to some degree, select the right time to draw power from the grid. Our prototype system explores the idea of a laptop that can take current electricity prices into account in determining when to charge its battery to full capacity. Laptops already include very sophisticated power management capabilities, but none of the current systems exploit electricity prices as part of their management strategies. Other types of appliances with storage are likely in the future, such as pluggable hybrid automobiles. Moreover, there are deployments of storage systems like flywheels that enable both backup electricity generation and an ability to select when power is taken from the grid [6].

In many cases appliances do not have sufficient embedded control to carry out their own demand response. This is clearly true of legacy appliances, which did not anticipate any such function, but also of devices, such as light bulbs, where costs and benefits are too low to merit individual embedded control. Even sophisticated devices may not provide demand response capabilities themselves because of the potential subtlety of these capabilities. Demand response algorithms are largely unexplored territory and can be expected to evolve, possibly into quite complex or coordinated strategies. These factors make it likely that many or most demand response actions will be controlled by a *unified hub* likely to be associated with the facility BAS. A unified hub provides control for a potentially large number of individual loads according to strategies that may be quite specific to the type of load. A general demand response architecture must therefore include the concept of a hub to control demand response for a range of facility loads. The hub and meter form the core of our demand response architecture.

Although there is a great deal that can be done with a facility hub supporting demand response and price-aware loads, there are limitations to basing demand response on the information and negotiating abilities at the level of a single facility. It is therefore important also to consider the idea of collections of facilities banding together and negotiating their own aggregated demand response strategies. Let us call the control center for such coalitions a *real-time aggregator*. This may be the utility itself. For instance, the ESP itself may offer a discount to users that will ensure that their loads are capped at a proposed level during peak periods. Or, a real-time aggregator could be an independent party who delivers a profile of demand response capabilities to the ESP. Real-time aggregators could be the solution to certain worrying possible consequences of basing demand response on per-facility control. For instance, there is a threat of *rebound* peaks in which facilities delay their demands to avoid a peak, but cause a new peak when trying to satisfy delayed demand [7]. A real-time aggregator may have the ability to schedule the satisfaction of delayed demand to avoid such rebound peaks. A general architecture for demand response should envision real-time aggregators and enable them to coexist with more localized capabilities. In particular, there must be a strategy for how real-time aggregators execute control on their loads in a way that makes sense even when unified hubs and independent smart appliances are in use.

3. Architecture

Let us assume that we are given an AMI that provides bidirectional information flows and, in particular, enables demand response based on real-time prices. Let us further assume that this capability is given to a facility that has BAS and/or a collection of intelligent appliances. Our goal is to construct an architecture that is capable of exploiting these assumptions to provide demand response that can realize incremental gains based on inexpensive modifications using control from any subset of potential control loci, including energy aggregators, BAS hubs, or intelligent appliances.

There are a variety of approaches to this problem based on different communication and control strategies. A simple approach is to let the MDMA advertise the prices on the Internet and let facilities subscribe to this information to plan demand response. This has some advantages, like simplicity and the exploitation of shared infrastructure, but it also has a variety of drawbacks. For instance, not all facilities will have the necessary Internet access, and there could be significant latency issues because of infrastructure sharing (so one facility may get its prices at a significantly different time from another). This problem can be addressed by the use of a network dedicated to delivering real-time pricing and designed to be respectful of uniform information distribution to its clients. Because of the growth of AMI, such a network is already available and not a prohibitive expense, so it is possible to use this, *provided* its facility-based node, the meter, is able to communicate directly with nodes in the facility. Our approach, the *Meter Gateway Architecture (MGA)* takes as its premise that the meter is the key communication link between the MDMA and the facility demand response controls. Given this premise, there are a few more issues to resolve concerning how communication and control are organized around the meter. The main additional element of the MGA is a *unified hub* that provides central communication and control within the facility and, in particular, is able to collect information from the facility meter and make it available thoughout the facility. ZigBee is a rapidly emerging technology for wireless communication in facilities, and it is ideal for MGA deployment of this hub.

The types of control in MGA are shown in Figure 1. L1 is a “smart” appliance that can provide its own demand response. An example is a dishwasher with an economy delay cycle that waits for cheap power. Here the demand response control C1 for the appliance

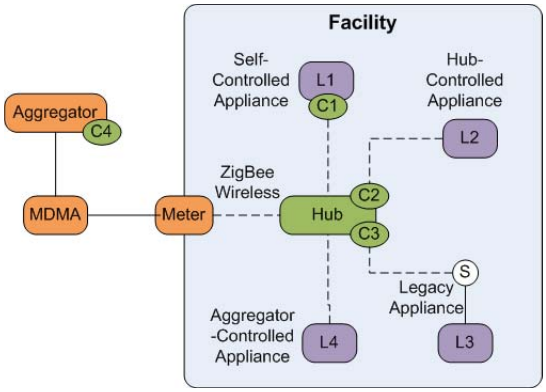


Figure 1. Types of Control in MGA

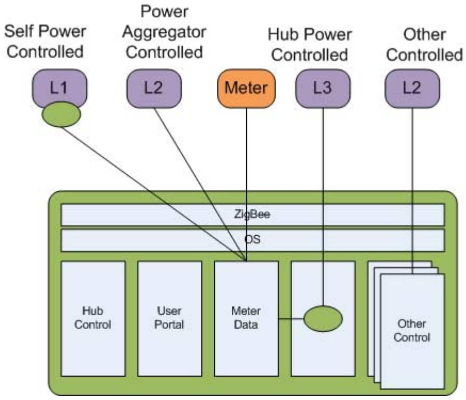


Figure 2. Hub Architecture Supporting MGA

is located in the embedded control of the appliance, which makes use of a ZigBee wireless link to the hub, where it gets real-time prices. By contrast, L2 is an appliance that delegates its demand response controls to the hub, which has the control functions C2 for executing demand response actions for L2. This may consist of putting L2 into a power-save mode at the right time. An example is a control for a refrigeration system that tries to optimize the cooling cycles of the refrigerator by varying its cooling range. Here the logic may better reside on the hub because the function is more complex than the appliance wishes to handle itself in its own (simple) embedded control. A legacy appliance L3 that has no embedded control of its own can be hooked to a switch that has control C3 at the hub. This has the disadvantage that it cannot make use of any intelligence in the appliance, but, since many legacy appliances have no intelligence anyway, there may be no major loss. Inexpensive loads like light bulbs will often also be in this category. Finally, L4 is an appliance controlled by a real-time aggregator. An example is an ESP that has direct controls over electrical water heaters and uses this control to disable water heating in emergency situations. Another is a collection of building owners that garner a good price from the ESP by enabling a power save mode that reduces load by 15% when requested by the ESP. In this case control C4 resides at the hub, and, following the MGA concept, the control is relayed to the participants using the MDMA. The real-time aggregator may be linked to the MDMA over the Internet or other connection, and the meter is linked to the MDMA by the communication channel provided by the AMI.

The most important elements of MGA are the meter and the hub. We assume the meter is able to get real-time prices from the MDMA and relay them to the hub over ZigBee. The architecture of a unified hub that supports MGA is shown in Figure 2. The device has a ZigBee interface and an Operating System (OS) capable of supporting some partitioning of processes. There are three distinguished processes. The *Hub Control* process provides basic hub services like authorizing and managing the installation and removal of control processes for appliances. The *User Portal* provides an interface for the facility owner, such as the ability to inspect hub values from a PC. The *Meter Data* process provides the price information that the MDMA supplies to the hub, making this available to appliances with embedded control like L1. It acts as a local agent for the control a real-time aggregator sends through the meter to L2. And, finally, it acts as a data source for other control processes. The unified hub is a general concept and there is a possibility for it to be used by a collection of applications different from demand response, so we envision it having any number of isolated controls for assorted sensors and actuators in the facility. The unified hub may be an independent element, or it could be integrated with the meter itself, or it could be integrated with a BAS control element. The main requirement is that it provide support for hub control, user portal, and meter data as illustrated in Figure 2.

This architecture is feasible to build. A meter-integrated version of the unified hub was defined in the attested meter [8], which focused on security features such as remote attestation.

4. Experiments

We now describe our MGA prototype and some experiments that illustrate its functionality. The test bed is illustrated in Figure 3. It contains five main components. The central component is a server computer

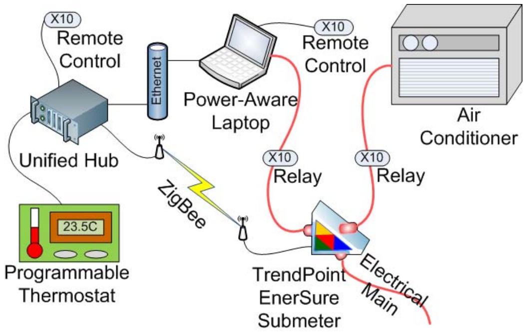
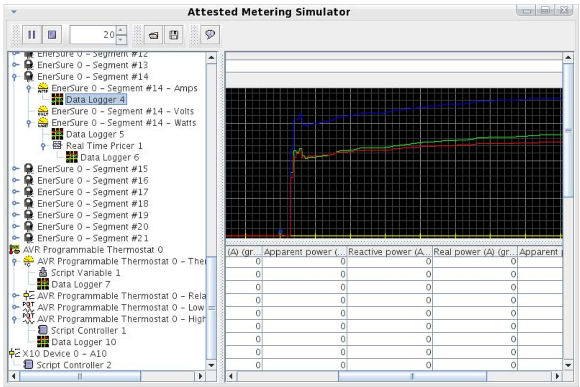


Figure 3. Test Bed

representing our unified hub. It controls and monitors the rest of the system using custom Java software installed on the computer. A USB-connected X10 remote control is attached to the unified hub to actually switch an air conditioner on and off. The second component is a laptop that represents an intelligent appliance capable of performing autonomous demand response based upon real-time prices provided by the unified hub over an Ethernet link. It also runs a Java application to actually implement the controls, and controls its own connectivity to the power mains using a second USB-connected X10 remote control that sends commands to a simple "appliance module" interposed between a power outlet and its power supply (Figure 4c). Power management was disabled on the laptop for this experiment to provide a more predictable power consumption profile, and it did not run any intensive applications during the experiments. The third component in the apparatus is a permanently-installed window air conditioner that is also controlled using an X10 module, although this module was custom-built since the three-phase wiring of the test apartment precluded the use of a standard 220V X10 control module. We use a TrendPoint EnerSure electrical submeter to measure the power consumption of the air conditioner and the laptop computer (Figure 4d). It is a Modbus device that is connected to the unified hub using an 802.15.4 wireless link (Figure 4b). 802.15.4 is the MAC layer in the ZigBee network protocol stack, so we refer to 802.15.4 network connections as ZigBee connections in the remainder of this section. Finally, we developed a custom digital programmable thermostat that is connected to the unified hub using a serial link so that the hub can adjust its high and low setpoints in response to price changes (Figure 4a). The unified hub is responsible for issuing actual control commands to the air conditioner module, due to a physical limitation of the X10 controller attached to the thermostat, but the unified hub relies on the thermostat to determine when the control commands should be issued. In a commercial implementation it is likely that the thermostat and the hub would be merged into a single unit.



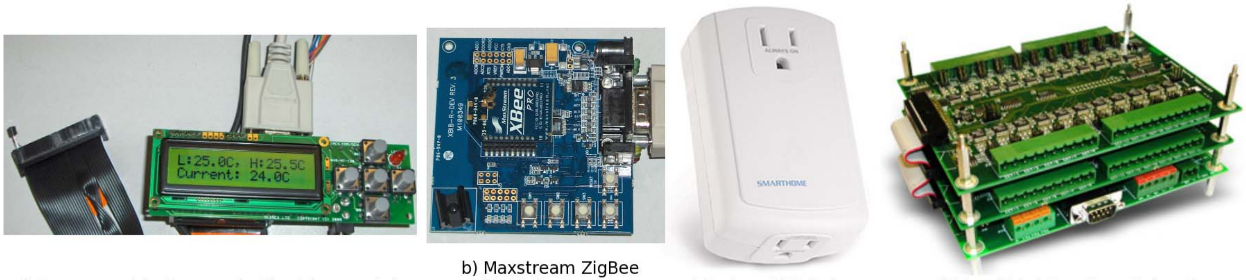
(A) (gr...	Apparent power (...	Reactive power (A...	Real power (A) (gr...	Apparent
0	0	0	0	
0	0	0	0	
0	0	0	0	
0	0	0	0	
0	0	0	0	
0	0	0	0	
0	0	0	0	
0	0	0	0	
0	0	0	0	

Figure 5. Test Bed Display of Air Conditioner Power Requirements Over Time

The Java testbench software used to control the experiment from the two computers is depicted in Figure 5. The left pane contains icons repesenting each device in the system. The right pane contains a graphical component that can be used to configure or monitor the currently selected device in the left pane. The software breaks each physical device represented in the left pane into primitive components that can be handled similarly by the software, even if the physical devices are quite different. For example, a single software component is used to configure the setpoints on the programmable thermostat and X10 lamp dimmer modules, since they are both discretely variable controls with a wide range of possible settings.

The setting of any primitive component can be recorded over time using a generic data logger component, as pictured in the right pane of Figure 5. This component attaches itself as a "decorator" to the component being logged and can maintain both a memory buffer of a limited number of readings as well as an on-disk file where readings are archived when the memory buffer overflows or the recording process is stopped. The archiver can use a standard Comma-Separated Value (CSV) format for easy import into spreadsheets for analysis, or a more compact binary format. We archive measurements of the following values with the indicated periods:

1) Ambient temperature of the apartment (*period: as short as possible, typically 1 second*).
2) High and low setpoints of the thermostat (*period: as short as possible, typically 1 second*).



a) Programmable Communicating Thermostat
b) Maxstream ZigBee RF Modem
c) Insteon X10 Relay
d) TrendPoint EnerSure Submeter

Figure 4. Test Bed Equipment

3) Current electricity price (*period: 1 hour*).
4) Power consumption of the two appliances under consideration (*period: 3-4 seconds*).
5) Individual bills for each appliance (*period: 3-4 seconds*).
6) Current charge percentage of the laptop's battery (*period: 10 seconds*).
7) States of the X10 relays for the laptop and air conditioner (*aperiodic*).

This information was all logged in CSV format for easy analysis using spreadsheet tools.

Every control supported in the Java testbench can be controlled manually or via a functional scripting engine. To use scripting, it is necessary to first define variables for each value that will be used in the script to determine control settings. To do this, additional decorator components are attached to the desired components in the testbench, and a variable name for each component is assigned. Our controls are dependent upon the current price of electricity and the charge capacity of the laptop battery, so it was necessary to define variables for those two values. Additionally, the thermostat's high setpoint was dependent on its low setpoint, so a variable was defined for the low setpoint. Finally, the X10 relay controlling the air conditioner simply imitates the actions of a relay on the programmable thermostat, so a variable was defined for the thermostat's relay.

Using additional decorator components, we can define an expression that will be periodically evaluated to determine a new value for each control that is to be scripted. We chose Scheme as our scripting language because a complete Scheme interpreter has been implemented in Java and can be easily embedded in Java programs [9]. We simply create a "let" expression associating the current value of each system element being watched with the appropriate variable name, and insert the control value expression for the element being updated in the surrounding expression. We then evaluate the expression and use the result as the new setting for the control. Scheme is a very flexible higher-order language, so sophisticated scripts can be developed within this environment.

The test bed is installed in an active single-occupancy apartment with approximately 500 square feet of air-conditioned floorspace. The apartment is on the north-facing side of a multi-story apartment complex and is a well-insulated unit constructed in the late 1990s. The apartment is located in Urbana, Illinois, and is served by AmerenIP. It is enrolled in the Power Smart Pricing plan offered by Ameren (the parent company of AmerenIP), which provides RTP support using day-ahead prices in the residential market. However, our experiments aim to suggest that hourly wholesale prices can be used directly to achieve economic benefits for an individual consumer, so we did not use the day-ahead prices in our experiments. Rather, we used historical hourly prices provided on Ameren's public website to guide our controls. Since we are experimenting with an air conditioning system, we selected prices from days that were as similar as possible to those on which we performed the experiments. We compute a bill for the electricity usage of the appliances being metered in our experiments using those prices. We compare the two bills and estimate the comfort level of the occupant throughout the trial period using recorded temperature measurements to evaluate the general effectiveness of our demand response system. Note, however, that we are not attempting to prove that the particular demand response algorithms we use are optimal or even suitable for such a task. The works cited earlier demonstrate the usefulness of demand response in general, and these experiments demonstrate that MGA is suitable for implementing demand response systems in a flexible manner.

In the first experiment, we allow the air conditioner (under the control of the unified hub) and power-aware laptop to optimize their operation in response to real-time prices. We started this experiment at midnight (00:00) on June 11, 2007, and completed it at eleven o'clock in the evening of the same day. June 11 and 12 both had high temperatures of 30.6 deg. C. Thus, we selected real-time prices from May 24, which reached a high temperature of 31.1 deg. C. The average rate of electricity on that day was 5.332 cents per kWh. To initialize the experiment, we allowed the air conditioner to adjust the ambient temperature of the apartment to a fixed setpoint of 24 deg. C before midnight.

We define four different expressions for the controls in our system. Some of them are dependent upon a value derived from the current and upcoming prices of electricity to affect the state of loads. This value is calculated to be the current price of electricity minus 50% of the difference between the price of electricity one hour from the current time and the current price of electricity. If that price is not available at the time of the calculation, a price forecast based upon past prices can be substituted, or it can be set to 0 to eliminate its effect. In the following discussion we refer to this value as the "effective price of electricity," which can be mathematically defined as follows:

$$P = P_n - (P_n - P_{n+1}) * 0.5,$$

where P is the "effective price of electricity," P_n is the current price of electricity, and P_{n+1} is the price of electricity during the next unit of time (hour, in this case). The effect of this adjusted price value is to make electricity relatively cheaper when a higher-price period is approaching, and relatively more expensive when a lower-price period is approaching.

In the first control expression, we allow the laptop's battery to run 10% lower before starting to recharge itself for every 1 cent increase in the effective price of electricity beyond a nominal low price threshold of five cents, with a minimum threshold of 50%. We then allow the battery to remain connected to AC power until its capacity is 10% above that threshold. The threshold is actually continuously variable, and does not typically change in increments of 10%.

The second and third controls apply to the air conditioner thermostat setpoints. We define a maximum low setpoint of 24.5 deg. C., which maintains an actual ambient temperature below 27 deg. C as will be shown later. So as not to negatively affect the occupant's comfort level, we will show that we can maintain a low average temperature in the apartment while still achieving a low average price per kWh by lowering both setpoints during low-price periods and slightly increasing both setpoints during peak periods. We define a nominal high-cost threshold of 12 cents per kWh, and decrease both setpoints from their maximum values by 0.5 deg. C for every 2 cent decrease in the cost of electricity. We round the resultant setpoint to the nearest half deg. C, to correspond to the precision of the thermostat. Mathematically, disregarding issues of precision, we can represent the low setpoint as follows:

$$S = 24.5 - \mathit{Max}(0, (0.12 - P) * 25),$$

where S is the low setpoint, P is the effective price of electricity in dollars per kWh, and the function *Max* returns whichever of its arguments is the greatest. The high setpoint is always 2 deg. C greater than the low setpoint.

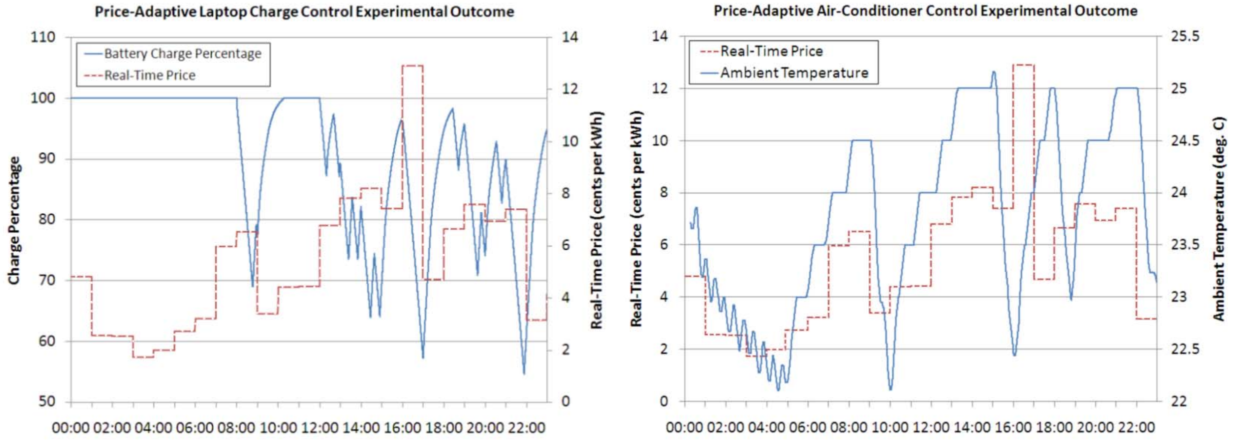
Finally, we must define a control expression to link the X10 relay with the thermostat's relay. We have assigned the thermostat's relay state to the variable `relayOn`, so we simply specify that variable name as the expression to be evaluated.

To provide baseline data against which to compare the results of our first experiment, we initialized the apartment under consideration to 24 deg. C and then re-ran the experiment outlined above on June 12. We set the low setpoint of the thermostat to 23 deg. C, and the high setpoint to 25 deg. C, which together allow the ambient temperature in the apartment to range between 22.5 deg. C and 25.5 deg. C. We also configured the laptop to simply keep its battery fully charged, like a standard laptop.

5. Evaluation and Implications of Results

The experiments described above completed successfully and produced data supporting the claim that the meter gateway architecture is a flexible demand response paradigm that supports multiple loci of control. In this section, we describe the results and their implications.

The results of the price-adaptive controls for a laptop battery on its charge capacity over time are presented in Figure 6a. It is obvious that the laptop relied more heavily on its battery during times of high prices than during low-price periods. The effects of taking future prices into consideration were also dramatic and can be observed clearly at 08:00, when prices were relatively low but still much higher than they were one hour in the future, and at 15:00, when prices were relatively higher but still extremely low compared to prices one hour in the future. The laptop consumed a total of 783 watthours, which cost 3.768 cents. In contrast, during the second experiment the laptop consumed only 763 watthours, indicating a slight inefficiency



(a) (b)

Figure 6. Experimental results

in the charge and discharge process, but incurred an electricity cost of 4.112 cents. Thus, using its own embedded intelligence, the laptop responded to RTP signals to achieve a cost reduction of 8.4% without ever draining its battery below 54% of its full capacity, although the controls did cause a slight increase in power consumption.

The results of the price-adaptive controls for an air conditioner on the ambient temperature of the apartment in the first experiment are presented in Figure 6b. A moving average of the ambient temperature with a window size of 17 minutes is used to reduce the artifacts of boundary behaviors in the digital thermostat. The oscillations between 00:00 and 05:00 are due to the air conditioner's internal thermostat, which causes it to stop operating for short periods when the air in its immediate vicinity drops below a certain minimum temperature. It is apparent that the differential of the curve is directly proportional to the effective price of electricity at most points. Periods of temperature reduction correspond to periods of air conditioner operation, so this indicates that the air conditioner did in fact operate during periods with relatively low effective prices, and rested during periods of relatively high effective prices. This demonstrates that the MGA can be effectively adapted to control legacy appliances using retrofit controllers and a central point of control we call the unified hub.

Furthermore, the results suggest that price-adaptive controls can lead to a decrease in the average price of the electricity consumed by the air conditioner. During the first experiment, the air conditioner consumed a total of 2573 watthours and cost 11.45 cents to operate, while maintaining an average ambient temperature of 23.13 deg. C. On the other hand, during the second experiment the air conditioner consumed only 1781 watthours, but cost 9.41 cents to operate and maintained an average temperature of 23.99 deg. C. Although using fixed setpoints led to a 31.8% reduction in energy consumption, it only reduced costs by 17.8%.

6. Related Work

Various types of demand-response programs have been developed over the past decades. Generally, they have achieved little success [10]. This is due to several reasons. First, demand response participants have typically relied upon manual response strategies rather than using automation, although automated response technologies are slowly becoming more prominent, particularly in industrial and commercial buildings [11]. Manual strategies are difficult to maintain due to significant volatility in real-time prices, requiring continual strategy adjustments. Programs have also been poorly advertised and promoted, and insufficient effort has been allocated to customer education [12]. That study also reports that the only demand response participants that have consistently provided significant load reductions are those with backup generators and those enrolled in demand response programs that impose mandatory load curtailment actions. However, mandatory programs tend to reduce program participation levels across all sectors, since customers ranging from home-owners to manufacturers all experience periods during which they are unable to tolerate service interruptions [13]. The architecture presented in this paper provides automation to reduce the burden on customers while still permitting them to have ultimate authority over their energy usage, unless they enroll in a mandatory program and those controls are disabled.

As discussed at the beginning of this paper, Advanced Metering Infrastructure (AMI) is likely to provide the communications infrastructure for future demand response projects. However, AMI systems are potentially subject to several security vulnerabilities that could impede their deployment or interfere with their operation. These vulnerabilities and a security architecture for mitigating them using virtualization and remote attestation are presented in [8].

7. Conclusion

We have presented the meter gateway architecture, which provides a general paradigm for automating demand response and other controls with support for multiple loci of control, ranging from intelligent appliances to remote parties on the other side of the meter from homes, businesses, and institutions. We evaluated the architecture using a prototype implementation based on reaily-available commercial components, and demonstrated its flexibility by simultaneously controlling a legacy appliance and an intelligent appliance in response to fluctuating real-time prices. The results of our experiments demonstrate the effectiveness of the architecture, and also suggest that automated demand response strategies will allow electricity customers to achieve cost savings using real-time prices that were previously unattainable using manual response strategies.

Acknowledgements

We would like to thank the TCIP Center team for their feedback on this work. This work was supported in part by NSF CNS05-5170 CNS05-09268 CNS05-24695, CNS07-16421, ONR N00014-04-1-0562 N00014-02-1-0715, DHS 2006-CS-001-000001 and a grant from the MacArthur Foundation. Michael LeMay was supported on an NDSEG fellowship from the AFOSR. The views expressed are those of the authors only. TrendPoint graciously donated to the Illinois Security Lab an EnerSure electric submetering device which was used in our experiments.

References

[1] S. Kiliccote, M. A. Piette, D. S. Watson, and G. Hughes, "Dynamic controls for energy efficiency and demand response: Framework concepts and a new construction study case in new york," in *ACEEE Summer Study on Energy Efficiency in Buildings*, Pacific Grove, CA, August 2006.

[2] M. Martinez and F. Russell, "Smart thermostats getting results at SCE," Utility Automation & Engineering, July 2007.

[3] D. Hammerstrom, "The grid friendly appliance controller," http://gridwise.pnl.gov/technologies/transactive_controls.stm.

[4] D. Egan, "The emergence of ZigBee in building automation and industrial control," *Computing & Control Engineering Journal*, vol. 16, no. 2, pp. 14–19, 2005.

[5] A. Goodloe, M. McDougall, R. Alur, and C. A. Gunter, "Predictable programs in barcodes," in *International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES '02)*. Grenoble, France: ACM, October 2002, pp. 298–303.

[6] I. Iglesias, L. García-Tabarés, A. Agudo, I. Cruz, and L. Arribas, "Design and simulation of a stand-alone wind-diesel generator with a flywheel energy storage system to supply the required active and reactive power," *IEEE Power Eng. Syst. Conf*, pp. 1381–1386, 2000.

[7] G. Heffner and D. Kaufman, "Distribution substation load impacts of residential air conditioner load control." *IEEE TRANS. POWER APPAR. SYST.*, vol. 104, no. 7, pp. 1602–1608, 1985.

[8] M. LeMay, G. Gross, C. A. Gunter, and S. Garg, "Unified architecture for large-scale attested metering," in *Hawaii International Conference on System Sciences*. Big Island, Hawaii: ACM, January 2007.

[9] S. Miller, "SISC: A complete scheme interpreter in java," Tech. Rep., 2003.

[10] J. D. Kueck and B. J. Kirby, "Demand response research plan to reflect the needs of the california independent system operator," Consortium for Electric Reliability Technology Solutions, Tech. Rep., 2004.

[11] Neenan Associates, Ernest Orlando Lawrence Berkeley National Laboratory, and Pacific Northwest National Laboratory, "How and why customers respond to electricity price variability: A study of NYISO and NYSERDA 2002 PRL program performance," New York State Energy Research and Development Authority, Tech. Rep., 2003.

[12] G. Barbose, C. Goldman, and B. Neenan, "A survey of utility experience with real time pricing," Ernest Orlando Lawrence Berkeley National Laboratory, Tech. Rep., 2004.

[13] B. J. Kirby, "Spinning reserve from responsive loads," Oak Ridge National Laboratory, Tech. Rep., 2003.