Residential Demand Response under Uncertainty

Paul Scott[13], Sylvie Thiébaux[13], Menkes van den Briel[13], and Pascal Van Hentenryck[23]

[1] Australian National University, ACT, Australia
[2] University of Melbourne, VIC, Australia
[3] NICTA, Australia
{first.lastname}@nicta.com.au

Abstract. This paper considers a residential market with real-time electricity pricing and flexible electricity consumption profiles for customers. Such a market raises an optimisation problem for home automation systems where they need to schedule consumption activities to reduce costs, whilst maintaining a base level of comfort and convenience. This optimisation problem faces uncertainty in real-time prices, weather conditions, and occupant behaviour. The paper presents two online stochastic combinatorial optimisation algorithms that produce fast, high-quality solutions to this problem. These algorithms are compared with reactive control strategies and a clairvoyant controller. Our results demonstrate the value of stochastic information and online stochastic optimisation in residential demand response.

1 Introduction

Electricity consumption in residential markets will undergo fundamental changes in the next decade due to the availability of solar panels and novel pricing mechanisms, progress in batteries and electric cars, and the emergence of smart appliances and home automation. These technologies provide residential customers with the ability to actively participate in smart grid activities such as demand response where loads are shifted to times favourable for the network as a whole.

Having an intelligent Home Automation System (HAS) within each home is a key component in this vision. The HAS receives information about device operating characteristics, usage requests and network signals, and can send control requests back to smart devices. The HAS provides occupants with feedback on their consumption habits and, more importantly, can make control decisions for itself. This control can be used to meet one or more of the following objectives:

1. Improve occupant comfort,
2. Reduce overall electricity consumption,
3. Perform demand response for network.

These objectives are often conflicting, so occupants need to indicate how they value comfort against cost savings in order to get the right balance for them overall. The task of the HAS is to decide on a series of control actions to take over time, which produces an optimal solution to the combined objectives. The HAS can implement simple policies to attempt to meet these conflicting objectives. Or, more interestingly, it can use sophisticated stochastic optimisation technology which exploits forecasts and observed patterns in prices, weather, residential activities and smart device usage.

This paper aims to determine the benefits of online stochastic optimisation for a HAS that is exposed to Real-Time Pricing (RTP) as a demand response mechanism. A number of research projects have started examining this very issue (see the related work section) but they often give an incomplete picture of the benefits of optimisation and the value of stochastic information. These projects often consider simpler uncertainty models, which give a partial understanding of the true benefits that optimisation can bring to this setting. In contrast, this paper makes two primary contributions: one conceptual and one algorithmic.

At the conceptual level, the paper presents a compositional architecture for HAS optimisation, where each device can be modelled independently in terms of a collection of functions that encapsulate its behaviour. These devices are then assembled into a model of a home, from which optimisation problems for the HAS can derive.

At the algorithmic level, the paper presents a comprehensive study of the value of HAS optimisation in the presence of uncertainty about future prices, occupant behaviour and environmental conditions. Our formulation uses models representative of physical devices and stochastic models trained on real weather and network demand data. These device and stochastic models are used in two online stochastic optimisation algorithms which are compared to simple control systems based on reactive policies.

The experimental results not only show the value of stochastic information, but also that stochastic optimisation provides solutions that are close to the clairvoyant solutions which have perfect knowledge of the future. The online stochastic algorithms using MILP technology are fast and produce significantly better solutions than the reactive controllers. Also of interest is the comparison between the two online stochastic algorithms, and an experiment that investigates the optimal rolling horizon duration.

The rest of the paper presents the deterministic HAS optimisation problem, its stochastic version, the stochastic models and finally the experimental results.

2 Deterministic HAS Optimisation

A house contains a collection of controllable devices which influence the amount of power consumed in the house and the level of comfort that residents experience. We consider the operation of these devices over discrete time steps[4]: $\forall i \in \mathbb{Z} : t_i \in \mathbb{R}$ where $t_i > t_{i-1}$ and $\forall i \in \mathbb{Z} : t_i^{stp} = t_i - t_{i-1}$.

Given a real time price for electricity and other input parameters (e.g., external temperatures and device requests), optimal operation of these devices is

[4] Variable time step sizes will be used to focus computational time where most needed.

achieved by minimising the sum of monetary and comfort costs. The optimisation decision variables are the device actions at each time step, which are constrained by device characteristics and total power limits on the house.

2.1 Formal Definition

We start with a new formal definition of a device, which is a collection of functions that govern the device operation. These include functions for permissible device actions, state updates, electrical power consumption/generation, and any non-power-related operation costs. The operation costs are always positive and may include any occupant comfort costs, fuel consumption or wear and tear on the equipment. By convention power consumed by the device is negative and power generated (e.g., by a rooftop photovoltaic system) is positive.

Definition 1 (Device). *A device is a tuple* $d = (A_d, S_d, R_d, q_d, g_d, f_d, l_d)$*, where:*

- $A_d \subseteq \mathbb{R}^{m_d} \times \mathbb{Z}^{m'_d}$ *is the set of device actions*
- $S_d \subseteq \mathbb{R}^{k_d} \times \mathbb{Z}^{k'_d}$ *is the set of device states*
- $R_d \subseteq \mathbb{R}^{w_d} \times \mathbb{Z}^{w'_d}$ *is the set of device input parameters*
- $q_d : S_d \times R_d \longrightarrow \mathcal{P}(A_d)$ *is the permissible action function*
- $g_d : A_d \times S_d \times R_d \longrightarrow S_d$ *is the state update function*
- $f_d : A_d \times S_d \longrightarrow \mathbb{R}$ *is the electrical power function*
- $l_d : A_d \times S_d \times R_d \times \mathbb{R} \longrightarrow \mathbb{R}$ *is the operational cost function*

A house is simply a set of devices, together with bounds on the instantaneous amount of power the house can transfer to or from the grid:

Definition 2 (House). *A house is a tuple* $h = (D_h, \underline{p}_h, \bar{p}_h)$*, where:*

- D_h *is the set of devices*
- $\underline{p}_h, \bar{p}_h \in \mathbb{R}$ *are the lower and upper power limits*

We now turn to the deterministic formulation of the HAS optimisation problem which will be later used as a building block for our stochastic formulation. The deterministic formulation assumes that the input parameters are known over a horizon of n time steps. The objective is to choose device actions to reduce the total cost over the horizon, which includes device operational costs and monetary costs from trading power with the network.[5] Inputs include the device initial states, the RTP, the house background power usage[6] and the device input parameters at each time step. The variables at each time step include the device actions and states, and the device and house power consumptions and costs.

We use the following notation: $(a)_+ = |a|$ if $a > 0$ and 0 otherwise, and similarly $(a)_- = |a|$ if $a < 0$ and 0 otherwise, where $a \in \mathbb{R}$.

[5] The RTP for net consumption or generation can be different.

[6] This aggregates uncontrollable electrical consumption, e.g., lighting, entertainment and cooking.

Definition 3 (Deterministic HAS Optimisation Problem).
For a house $h = (D_h, \underline{p}_h, \bar{p}_h)$, the HAS optimisation problem over a horizon of $n \in \mathbb{N}^*$ time steps is the following:

Inputs:
for each device $d = (A_d, S_d, R_d, q_d, g_d, f_d, l_d) \in D_h$
- $s_{d,0} \in S_d$ *is the device initial state*

for each device $d \in D_h$ and time step $i \in \{1 \ldots n\}$
- $r_{d,i} \in R_d$ *are the device input parameters*

for each time step $i \in \{1 \ldots n\}$
- $p^b_{h,i} \in \mathbb{R}_-$ *is the house background power*
- $v_i \in \mathbb{R}^2$ *is the real-time price (buying, selling)*

Decision variables:
for each device $d \in D_h$ and time step $i \in \{1 \ldots n\}$
- $a_{d,i} \in A_d$ *are the device action variables*

Other variables:
for each device $d \in D_h$ and time step $i \in \{1 \ldots n\}$
- $s_{d,i} \in S_d$ *are the device state variables*
- $p_{d,i} \in \mathbb{R}$ *is the device power*
- $c_{d,i} \in \mathbb{R}_+$ *is the device operation cost*

for each time step $i \in \{1 \ldots n\}$
- $p_{h,i} \in [\underline{p}_h, \bar{p}_h]$ *is the total power*
- $c_{h,i} \in \mathbb{R}$ *is the total cost*

Constraints:
for each device $d \in D_h$ and time step $i \in \{1 \ldots n\}$
- $a_{d,i} \in q_d(s_{d,i-1}, r_{d,i})$ *is the action permissibility constraint*
- $s_{d,i} = g_d(a_{d,i}, s_{d,i-1}, r_{d,i})$ *is the state update constraint*
- $p_{d,i} = f_d(a_{d,i}, s_{d,i})$ *is the device power constraint*
- $c_{d,i} = l_d(a_{d,i}, s_{d,i}, r_{d,i}, t_i^{stp})$ *is the device cost constraint*

for each time step $i \in \{1 \ldots n\}$
- $p_{h,i} = \sum_{d \in D_h} p_{d,i} + p^b_{h,i}$ *is the house power constraint*
- $\underline{p}_h \leq p_{h,i} \leq \bar{p}_h$ *is the house power limits constraint*
- $c_{h,i} = \sum_{d \in D_h} c_{d,i} + t_i^{stp} v_{i,1}(p_{h,i})_- - t_i^{stp} v_{i,2}(p_{h,i})_+$ *is the house cost constraint*

Objective:
$\min \sum_{i=1}^{n} c_{h,i}$

2.2 Modelled Devices

In our experiments we consider a modern house with electrical HVAC, hot water heating, solar panels, a washing machine, a clothes dryer and a dish washer. We also include two devices that are expected to become popular within the next decades: an electric vehicle (EV) and a dedicated battery for storing electrical energy. Descriptions of these devices are given in this section. Some liberty has been used in these descriptions to aid understanding, however note that with slight reformulation they all fit into the rigorous device definition of the previous section. Device electrical powers and operational costs are consistently represented by the variables p_i and c_i, and power consumed by a device takes on a negative number.

The physical behaviour of devices has been approximated by linearising their physical equations and discretising time. Only significant steps of this process are mentioned in the device descriptions. For the experiments parameters were selected to be representative of typical devices. For example, the EV battery capacity is equivalent to that of a Nissan Leaf, and the house floor area for heating purposes is typical of an average-sized house. Some parameters were difficult to source so had to be estimated such as the efficiency of the EV battery.

Battery. A battery has a stored energy state $E \in [0, \bar{E}]$ and a charge/discharge power $p \in [\underline{p}, \bar{p}]$ action variable. The battery has a fixed efficiency $\eta \in [0, 1]$. The stored energy state update function is given by:

$$E_i = E_{i-1} + t_i^{stp} \left(\eta (p_i)_- - (p_i)_+ \right) \tag{1}$$

A battery lifetime cost c is associated with power that is discharged from the battery through a lifetime price v: $c_i = v(p_i)_+$.

Electric Vehicle. An electric vehicle (EV) is like the battery above, but with a few additional constraints. Firstly the input parameter $x^h \in \{0, 1\}$ indicates whether the EV is home, and the battery can only be charged/discharged when this is the case:

$$x_i^h = 0 \implies p_i = 0 \tag{2}$$

The input parameter $p^d \in \mathbb{R}_+$ represents the power drawn from the battery whilst driving. This modifies the state update function as follows:

$$E_i = E_{i-1} + t_i^{stp} \left(\eta (p_i)_- - (p_i)_+ - p_i^d \right) \tag{3}$$

The final constraint is on the amount of energy stored in the battery. The house occupants provide an input parameter $E^m \in [0, \bar{E}]$ that represents the minimum energy that the EV battery should have in it at each time. This value represents how much energy the occupant expects to need if they drive away in the car at a given time. This is not a hard constraint as the draw from driving can bring the battery charge below this limit, but it ensures that if the battery power does fall below, then it charges back up as fast as possible.

$$x_i^h = 1 \implies E_i \geq \min \left[E_{i-1} + t_i^{stp} \left(-\eta \underline{p} - p_i^d \right), E_i^m \right] \tag{4}$$

Hot Water Heating. The hot water system is made up of a storage tank and an electric heating element. We ignore the details of the interaction between hot and cold water in the tank and consider the state of the tank as being the amount of energy $E \in [0, \bar{E}]$ it contains above the inlet cold water temperature. The tank is considered empty of hot water when this value is zero. The action variable is the power setting of the electric heater $p \in [\underline{p}, 0]$ at each time step. The input parameter $p^d \in \mathbb{R}_+$ is the amount of power drawn from the tank to meet occupant demand. The energy state update function is given by:

$$E_i = E_{i-1} + t_i^{stp}\left(-p_i - p_i^d - p_i^l + p_i^u\right) \tag{5}$$

The variable $p^l \in \mathbb{R}_+$ represents thermal losses from the tank to the outdoor environment. The rate of loss depends on how full the tank is and the difference in temperature between the water set point $T^s \in \mathbb{R}$ and the outdoor temperature $T^o \in \mathbb{R}$ through a resistivity $R \in \mathbb{R}_+$:

$$p_i^l = \frac{1}{R}\frac{E_i}{\bar{E}}(T^s - T_i^o) \tag{6}$$

The variable $p^u \in \mathbb{R}_+$ is a recourse variable that is used to indicate the amount of hot water demand which goes unmet, i.e. water drawn from the tank when it is empty. This is heavily penalised as a cost c through an unmet demand price v: $c_i = vp_i^u$.

The hot water system has a minimum stored energy level $E^m \in [0, \bar{E}]$, much like the electric vehicle. If drawn water brings the energy level of the tank below this value then the heater must work as hard as possible to bring the energy back up. Occupants can adjust this input parameter to reduce the likelihood of running out of hot water.

$$E_i \geq \min\left[E_{i-1} + t_i^{stp}\left(-\underline{p} - p_i^d - p_i^l + p_i^u\right), E^m\right] \tag{7}$$

Under-Floor Heating/Cooling. The house temperature is controlled by a heat pump that heats/cools water, which is then pumped through piping embedded in the floor of the house. The temperatures of the floor and the air in the room $T^f, T^a \in \mathbb{R}$ are the device state variables. The action variable is the amount of thermal energy that is supplied to the floor of the house $p^t \in \mathbb{R}$. This is limited by the heat pump electrical power consumption $p \in [\underline{p}, 0]$ through heating and cooling Coefficients of Performance (COP) $\eta^h \in [\underline{\eta}^h, \bar{\eta}^h], \eta^c \in [\underline{\eta}^c, \bar{\eta}^c]$:

$$p_i = -\frac{1}{\eta_i^h}(p_i^t)_+ - \frac{1}{\eta_i^c}(p_i^t)_- \tag{8}$$

The COPs depend on the temperatures of the two thermal wells between which the heat pump is operating. We assume that the internal thermal well is at a constant temperature and that the external well is at the outdoor temperature $T^o \in \mathbb{R}$. We approximate the COPs as linear functions of T^o for some constants $a^h, a^c \in \mathbb{R}_+$ and $b^h, b^c \in \mathbb{R}$, with hard upper and lower limits:

$$\eta_i^h = \min\left[\max\left[a^h T_i^o + b^h, \underline{\eta}^h\right], \bar{\eta}^h\right], \quad \eta_i^c = \min\left[\max\left[-a^c T_i^o + b^c, \underline{\eta}^c\right], \bar{\eta}^c\right] \tag{9}$$

Heat can transfer between the floor and the outdoor environment $p^{fo} \in \mathbb{R}$, the floor and the air in the room $p^{fa} \in \mathbb{R}$, and the air in the room and the outdoor environment $p^{ao} \in \mathbb{R}$. We use simple lumped thermal resistivities $R^{fo}, R^{fa}, R^{ao} \in \mathbb{R}_+$ to govern these heat flows:

$$p_i^{fo} = \frac{1}{R^{fo}}(T_i^f - T_i^o), \quad p_i^{fa} = \frac{1}{R^{fa}}(T_i^f - T_i^a), \quad p_i^{ao} = \frac{1}{R^{ao}}(T_i^a - T_i^o) \tag{10}$$

The temperature state update functions are given by:

$$T_i^f = T_{i-1}^f + \frac{t_i^{stp}}{m^f \kappa^f} \left(p_i^t - p_i^{fo} - p_i^{fa} + A^f I_i \right) \tag{11}$$

$$T_i^a = T_{i-1}^a + \frac{t_i^{stp}}{m^a \kappa^a} \left(p_i^{fa} - p_i^{ao} + p_i^g \right) \tag{12}$$

where $m^f, m^a, \kappa^f, \kappa^a \in \mathbb{R}_+$ are the floor and air, mass and specific heat capacity coefficients respectively. Sunlight enters through the windows at an irradiance $I \in \mathbb{R}_+$ and lands on a floor area $A^f \in \mathbb{R}_+$. The input $p^g \in \mathbb{R}_+$ is thermal power generated by occupant metabolisms and background electric appliances which contributes to heating the air in the room.

The final relation we have is for the comfort cost c which depends on the distance of the air temperature from an occupant specified set point temperature $T^s \in \mathbb{R}$. The occupants also specify two time-varying comfort prices v^a, v^b, one of which is only included after a threshold temperature difference ΔT^{th}:

$$c_i = \begin{cases} v_i^a |T_i^a - T_i^s| & \text{if } |T_i^a - T_i^s| < \Delta T^{th} \\ (v_i^a + v_i^b)|T_i^a - T_i^s| & \text{otherwise} \end{cases} \tag{13}$$

Shiftable Loads. Shiftable loads are devices that need to run once within a time window. An occupant sets two input parameters: a start time i^s and a last allowed start time i^l, between which the controller must schedule the device to run. Examples of this kind of device include washing machines, clothes dryers and dish washers. We model non-preemptive shiftable loads which can have time varying power consumptions.

The start of run indicators $x \in \{0, 1\}$ act as both the device action and state variables. A shiftable load has a cumulative energy consumption function $\psi : \mathbb{R}_+ \longrightarrow \mathbb{R}_+$ which takes a run duration and returns the cumulative amount of energy that the device has consumed up to that duration. Constraints on the run indicator variables and the device power $p \in \mathbb{R}_-$ are given by:

$$\sum_{k=i^s}^{i^l} x_k = 1, \quad p_i = -\sum_{k=i^s}^{i} x_k \frac{\psi(t_i - t_{k-1}) - \psi(t_{i-1} - t_{k-1})}{t_i^{stp}} \tag{14}$$

Photovoltaics. The photovoltaic (PV) panels have no action variables, the amount of electricity they generate is purely determined by the solar irradiance input parameter. We model a PV system ignoring temperature and shading effects and by assuming the panels lay on a horizontal surface. The generated electric power $p \in \mathbb{R}_+$ is then a simple function of the panel area $A \in \mathbb{R}_+$, efficiency $\eta \in [0, 1]$ and global irradiance input parameter $I \in \mathbb{R}_+$: $p_i = \eta A I_i$.

3 Stochastic HAS Optimisation

So far we have considered the deterministic HAS formulation that requires perfect foresight about what will happen over the time horizon. In practice, almost all the input parameters are uncertain and their uncertainty is only revealed in real time (e.g., outdoor temperature) or in some cases a few time steps in advance (e.g., RTP). This motivates the use of online stochastic optimisation [1], which exploits statistical models of the uncertain parameters in order to make the best decisions on average.

3.1 The Stochastic Model

In the stochastic HAS problem the RTP v_i, background house power $p^b_{h,i}$ and device input parameters $r_{d,i}$ are random variables. We denote their real-world realisations (i.e. their values when the uncertainty is revealed) with the symbol *. For instance, T^{o*}_i denotes the real outdoor temperature at time step i. For notational convenience, all inputs are combined into one vector

$$z_i = (v_i, p^b_{i,h}, r_{d_1,i}, r_{d_2,i}, \ldots)^T \tag{15}$$

where we index elements with a k (e.g., $z_{i,k}$). Random variables at time step i may be dependent on each other and on the variables at previous time steps. Therefore the joint distribution for random variables up to time step i is given by:

$$P(z_i, z_{i-1}, \ldots) \tag{16}$$

Let t^* represent the current real world time. Each input $z_{i,k}$ is revealed a fixed amount of time $\Delta t^{rev}_k \in \mathbb{R}_+$ in advance (or in real time if $\Delta t^{rev}_k = 0$). This means, that for a given t^* an input $z_{i,k}$ is known to be $z^*_{i,k}$ if $t_i \leq t^* + \Delta t^{rev}_k$, otherwise it is a random variable. Given i and t^* we use $K_{i,t^*} = \{k | t_i \leq t^* + \Delta t^{rev}_k\}$ to denote the set of known input vector indices.

3.2 Online Stochastic Optimisation

In online stochastic optimisation decisions are made one step at a time using stochastic information about future events. After each time step the uncertainty and the effect of all actions is revealed, updating the state of the system. Decisions for the next period are computed and the process is repeated. It has been used successfully on a wide variety of problems (e.g., [2, 1]).

Our algorithms use a rolling finite horizon as illustrated in Fig. 1, where the time steps $1, \ldots, n$ are aligned to each horizon with $t_0 = t^*$. Optimisation is performed for each horizon using stochastic information for any unrevealed inputs and then the actions for the first time step are executed in the real world.

It might not be possible to execute actions produced by the optimisation if the real world input parameters z^*_1 differ from what the optimisation anticipated. For example, if the optimisation decides to run the hot water heater at full power

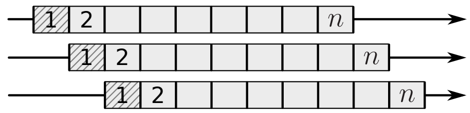


Fig. 1: Rolling horizon for 3 consecutive iterations.

and the tank unexpectedly reaches its capacity (due to less demand for hot water than expected), then the power of the heating action will need to be reduced so as to remain within the tank's capacity. Our HAS handles this automatically in the execution step, by using very simple executives for each device which select the closest feasible action.

The following sections introduce two approaches to solving the stochastic optimisation problem within each horizon: the expectation and the 2-stage algorithms.

3.3 Expectation Formulation

The expectation online stochastic algorithm takes the conditional expected value of any unrevealed inputs in the optimisation horizon, and solves the deterministic version of the problem given in Definition 3. We use the term expected value loosely because in truth we calculate the expected value only where it makes sense, which is typically for continuous inputs. For the rest of the inputs the most likely value is calculated instead. For example, expected value is used for outdoor temperatures and most likely value for the washing machine requests. Both of these calculations are performed using the joint distribution for inputs in the horizon, conditioned on any known inputs in and prior to the horizon:

$$P(z_n, z_{n-1}, \ldots, z_1 | (z^*_{n,k}, \forall k \in K_{n,t_0}), \ldots, (z^*_{1,k}, \forall k \in K_{1,t_0}), z^*_0, \ldots) \tag{17}$$

3.4 2-Stage Formulation

In this algorithm 2-stage stochastic programming is used within each horizon. This provides an approximation to a full multi-stage stochastic program which are, in general, known to be extremely challenging computationally [3]. The first stage includes time step 1, and the second stage time steps $2, \ldots, n$. Traditionally, in 2-stage stochastic programming there is no uncertainty in the first stage [4]. However, in our problem we are required to make decisions before all inputs in the first stage are revealed. To resolve this, first stage inputs are set to their real values if revealed, otherwise their conditional expected value is taken (as described in Section 3.3).

The second stage uses sampled scenarios to represent the uncertainty in the input parameters. We define a second stage scenario s as being a sample from the joint distribution of random variables in the second stage, conditioned on any revealed inputs in the second stage, and inputs in and prior to the first stage:

$$s \sim P(z_n, z_{n-1}, \ldots, z_2 | (z^*_{n,k}, \forall k \in K_{n,t_0}), \ldots, (z^*_{2,k}, \forall k \in K_{2,t_0}), \\ z_1, z^*_0, \ldots) \quad (18)$$

We use the Sample Average Approximation (SAA) [4] to limit the number of scenarios $S \in \mathbb{N}$ that we need to consider in the second stage. Each scenario in the second stage needs to have its own set of variables in the optimisation problem. For example we denote the power of device d at time step i in scenario s by $p_{d,i,s}$. The 2-stage objective function is given by:

$$\min \left[c_1 + \frac{1}{S} \sum_{s \in \{s_1, \ldots, s_S\}} \sum_{i=2}^{n} c_{i,s} \right] \quad (19)$$

3.5 Stochastic Inputs

Stochastic inputs include the real-time pricing (RTP), outdoor temperature, solar irradiance, background power, internal heat generation, hot water demand, EV usage and shiftable load requests. Accurately modelling any of these random processes is a significant undertaking in itself. The models we developed, while not the most sophisticated, suit the purposes of our experiments by capturing the fundamental nature of these stochastic processes. We investigated a number of different model types before settling on Generalised Additive Models (GAM) [5] for the continuous variables like temperature, and Markov Models for the more discrete occupant driven behaviours such as shiftable device requests.

Generalised Additive Models. In order to predict future values, the GAMs take advantage of weather forecasts that can be readily obtained from national weather services. These forecast values include daily maximum and minimum temperatures, as well as morning and afternoon cloud cover and wind speeds. The models also take in the value from the previous time step and temporal information. The models were trained on data obtained from the Bureau of Meteorology[7] and Australian Energy Market Operator[8] relevant to the states of NSW and the ACT in Australia.

The best way of implementing RTP in retail markets is still an open question and so is worth particular mention. It is unlikely that it will be a simple replication of the wholesale spot market price due to its high volatility. More likely it will be set by retailers, but it will have a shape that is representative of the wholesale market. We designed our RTP to be a quadratic function[9] of the amount of power that fossil fuel sources must supply to meet total network load. This is the total network demand minus the generation from renewable sources

[7] Bureau of Meteorology (BOM), www.bom.gov.au
[8] Australian Energy Market Operator (AEMO), www.aemo.com.au
[9] The quadratic is representative of an increasing marginal supply price [6].

such as wind and solar. We used a GAM for the total network demand and the generation from renewables is a function of wind speed and solar irradiance. The RTP is only revealed to a house 30 minutes in advance.

Markov Models. Semi-Markov models were used to capture the behaviour of four occupants of a specific house in the ACT. These models provide the consumption patterns and profiles for input parameters such as hot water demand, shiftable load requests and EV usage. Each model identifies the key activities of an occupant (e.g., sleeping, taking a shower and leaving for work), and specifies the probabilities of transiting from one activity to the next within certain time windows. Each activity is associated with a series of actions (e.g., watching TV and requesting the dish washer) that trigger changes in input parameters. Conditional sampling through these models is used to generate scenarios.

Whilst this scheme was convenient for our experiments, other more data-driven options are possible: we could simply gather and use a database of raw scenarios, or learn model parameters from disaggregated demand data [7, 8].

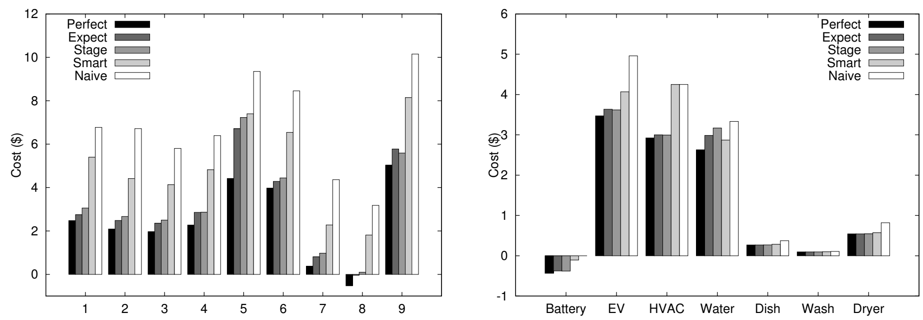
4 Experiments

We implemented the expectation and 2-stage online algorithms using Gurobi as a backend to solve the MILP within each horizon. The devices in Section 2.2 were implemented and included in the experimental house, and conditional samplers were created for the uncertain input parameters in Section 3.5. We created a simple simulator that uses the same physical equations as the optimisation to simulate the execution of actions in the real world. We compare the performance of the expectation and 2-stage controllers with naive and smart reactive controllers, and a controller that has perfect information.

The *Naive* reactive controller represents a household that either has no ability or no desire to respond to a RTP. It starts shiftable devices as soon as a request is received, fills up the hot water tank in off-peak hours, charges the EV only if it is below the requested minimum level, maintains the room at the set point temperature and never uses the battery bank.

The *Smart* reactive controller uses simple device action policies to decide how to respond to changes in RTP. It delays running a shiftable device until it reaches either a cheap price or the last available start time; uses thresholds about a moving average of the RTP to decide when to charge or discharge energy from the batteries, EV and hot water system; and maintains the room at the set point temperature like the naive controller.

The *Perfect* controller has perfect foresight about what will happen in the future. It optimises the deterministic problem in Definition 3 over the whole experiment duration with full knowledge of z_i^*. This controller (which is infeasible in practice) is used to give a lower bound on the objective that can be achieved by the other controllers.



(a) Controller comparison.

(b) Disaggregated costs.

Fig. 2: Costs for each experimental run and device costs averaged over runs.

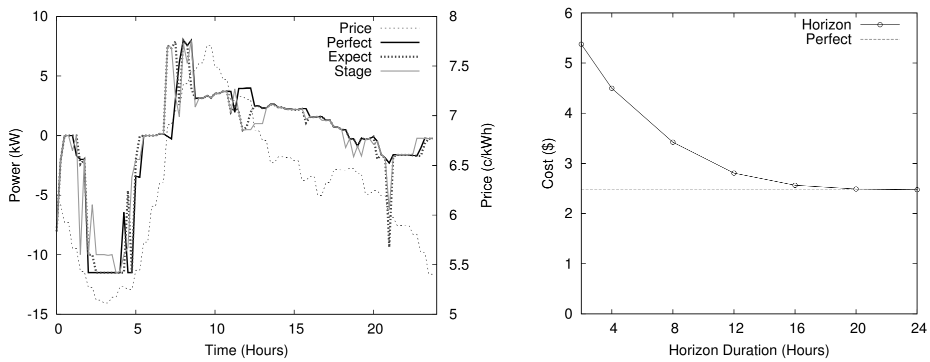
4.1 Controller Comparison

A total of 9 sets of input parameters typical for the month of February were generated. These were used in 9 separate experimental runs, each with a duration of 7 days. The online algorithms had 16 hour optimisation horizons, with 15 minutes for the first two time steps and 30 minute time steps for the remainder of the horizon.[10] The reactive and perfect controllers had 15 minute time steps. The 2-stage algorithm sampled 60 scenarios for each second stage. The amount of time spent optimising in Gurobi per day was on average 1 second for the expectation algorithm and 4 minutes for the 2-stage algorithm (using a single core of an Intel i7-2600 3.4GHz CPU). Whilst the 2-stage is much slower, its computational time can still be considered small when spread out over a day.

The controller costs are plotted in Fig. 2a for each experimental run. These results are adjusted to account for any energy that remains in the battery, EV, or hot water system at the end of an experimental run. This is done by valuing the left-over energy at the average RTP for the last 24 hours. Without this adjustment it would not be a fair comparison since any controller that anticipates the need to store energy for a future purpose would perform poorly if it does so just before the experiment ends. This is an artefact of the finite length of our experimental runs; with very long durations this problem goes away as the left-over energy costs become insignificant.

We see that the expectation and 2-stage algorithms get quite close to the performance of the controller with perfect foresight and they achieve significant cost reductions over the two reactive controllers ($\sim 35\%$ less than smart reactive controller). The expectation controller outperforms the 2-stage controller on average and in all runs except 9. This appears to suggest that the expectation controller is superior as it requires less computation and achieves lower costs. There are however a few subtleties to this that are worth discussing.

[10] By using larger time steps for more uncertain values further into the future we reduce the computational burden with only a minor reduction to solution quality.



(a) House power profiles.

(b) Perfect horizon.

Fig. 3: House power profiles for one day and experimenting with horizon lengths.

Fig. 2b shows the average split in costs between devices, ignoring the PV. The costs for the expectation and 2-stage controllers are essentially identical except when it comes to the hot water heater. The hot water heater is different from the other devices because it has a recourse variable for unmet hot water demand which has a very high cost associated with it. This recourse variable takes on a non-zero value in run 5 for the expectation and 2-stage controllers, and run 9 for the expectation controller, in all cases because the controllers fail to anticipate large spikes in hot water demand until it is too late. The reactive controllers have to be quite conservative with the amount of hot water they keep stored, so they never encounter a demand that they cannot meet (at the expense of higher prices paid for heating the water and greater thermal losses over time).

We in general expect the 2-stage algorithm to be more conservative than the expectation algorithm and to avoid having any unmet demand because through sampling it can identify upcoming peaks. This is the case in all runs except 5. With further investigation we found that in run 5 the 2-stage algorithm failed to generate the scenario with high demand when it was needed. The reason for this occurring was not due to too few second stage scenarios, but because of the way that second stage scenarios are conditionally sampled from the first stage which can take on expected values. In this instance, at a critical point in time, the first stage expected values precluded the high demand scenario from being able to occur in the second stage even though it was still physically possible.

What these results show is that the expectation algorithm typically does a very good job, but there are certain types of devices and random processes for which it performs poorly. We initially thought that the online 2-stage algorithm presented in this paper would be able to overcome these limitations, but there appears to be some problems associated with taking the expected value of first stage variables.

Fig. 3a gives an example of the power exchanged between the house and the grid for one day, along with the RTP. As expected, most consumption occurs when the price is low and when the price is high power is sold back to the grid from the battery, EV and PV. The expectation and 2-stage controllers follow the general trend of the perfect controller with some small divergences.

Fig. 3b shows the results of an experiment where we investigated how performance changes with the horizon duration. This plot shows the performance of the perfect controller running as an online algorithm where it is restricted to only having perfect foresight a certain distance into the future. The experiment is performed on run 1 for a number of different horizon durations and the results are compared to the original perfect controller that can see the full 7 days. The results show that there is little to be gained by looking any further into the future than 20 hours.

5 Related Work

Much of the existing literature on residential demand response focuses on deterministic formulations over fixed horizons where the scheduler has perfect foresight [6, 9]. Those that have considered uncertainty in the problem typically focus on just one aspect (e.g., real-time pricing) [10], or use very simple models for random variables [11].

Model-predictive control has been used to account for the uncertainty of estimated device model parameters and measurement noise [12], but not the uncertainty of the type we model. In general, model-predictive control is best suited to unconstrained, purely continuous settings with limited uncertainty.

Dynamic programming [11, 13] and Q-learning [14] have been used in conjunction with Markov Decision Process (MDP) formulations of the residential load scheduling problem, to generate policies that allocate power to each device. MDP approaches suffer from severe scalability issues, especially since the state space needs to be discretised. Moreover, MDPs seem somewhat excessive for our problem, given that uncertainty does not depend on the decisions taken. Our stochastic programming approach which uses scenario sampling is more scalable and more natural in the presence of exogenous uncertainty.

One paper [11] found that acting on the basis of the optimal dynamic programming solution did not provide any benefit over acting on expectations. For the most part we found this to be true, but as discussed in the experimental section we identified circumstances where the greedy nature of an expectation algorithm can lead to poor results. Our use of more complex uncertainty models and different devices is likely the reason why we had this extra observation.

The paper closest to ours compares two-stage stochastic programming and robust optimisation techniques for scheduling residential loads [15]. Uncertainty is restricted to the RTP which is known for the first stage but becomes uncertain thereafter. The objective includes minimising expected price and the probability mass of “risky” scenarios whose price exceeds a certain threshold. Comfort is handled by imposing hard constraints under which appliances must run, rather than by inclusion into the objective. In this setting, two-stage stochastic programming was observed to provide benefits over robust scheduling.

The scope of our analysis goes significantly beyond these results, by exploring uncertainty from a large range of sources and by identifying the value of stochastic information. We enable richer sources of uncertainty to be considered in our framework, by allowing inputs to be revealed at arbitrary points in time.

Commercially available residential DR solutions[11] typically focus on direct load control or simple reactive policies. Such systems could experience more optimal DR performance and greater residential customer satisfaction by using our algorithms.

6 Conclusion and Future Work

This paper contributes to the growing body of work on residential control of loads and storage under real-time pricing, by developing a framework that accounts for uncertainty. To our knowledge, it is the first work that provides a scalable and accurate solution in the presence of uncertainty about future prices, occupant behaviour and environmental conditions. Using models representative of physical devices and random processes, we have shown the monetary and comfort cost savings that can be achieved by using online stochastic algorithms over reactive control, and the comparison of performance between a 2-stage approach and acting on expectations. Studies such as the one in this paper are import for rallying industry and customers towards more effective energy management schemes.

Further research is needed to investigate how closely reality can be modelled with random processes, and if in turn they are suitable for online learning. We also need to further investigate how time step sizes and the number of second stage scenarios influence performance, and to conduct more experiments for different months of the year. The experimental set up we have developed can be used to experiment with and compare different pricing schemes. For example, time of use pricing and RTP where the price for generation is different from that for consumption. We also plan on investigating how multiple houses react to a RTP and what sort of emergent behaviour develops when they are all learning their statistical models online.

Acknowledgements. This work is supported by NICTA's Optimisation Research Group as part of the Future Energy Systems project. We thank our project members and reviewers for useful discussions and helpful suggestions. NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

[11] e.g., comverge: www.comverge.com, nest: www.nest.com and Cooper Power Systems: www.cooperindustries.com

References

1. Van Hentenryck, P., Bent, R.: Online Stochastic Combinatorial Optimization. The MIT Press, Cambridge, Mass. (2006)
2. Powell, W., Simao, H., Bouzaiene-Ayari, B.: Approximate dynamic programming in transportation and logistics: a unified framework. EURO Journal on Transportation and Logistics **1**(3) (2012) 237–284
3. Shapiro, A.: On Complexity of Multistage Stochastic Programs. Operations Research Letters **34**(1) (January 2006) 1–8
4. Shapiro, A., Dentcheva, D., Ruszczyński, A.: Lectures on Stochastic Programming: Modeling and Theory. MPS-SIAM series on optimization. Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104) (2009)
5. Hastie, T., Tibshirani, R.: Generalized Additive Models. Monographs on Statistics and Applied Probability Series. Chapman & Hall, CRC Press (1990)
6. Ramchurn, S.D., Vytelingum, P., Rogers, A., Jennings, N.R.: Agent-based control for decentralised demand side management in the smart grid. In Tumer, Yolum, Sonenberg, Stone, eds.: Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems - Innovative Applications Track (AAMAS 2011), Taipei, Taiwan (May 2011) 330–331
7. Kolter, J.Z., Batra, S., Ng, A.Y.: Energy disaggregation via discriminative sparse coding. In: 24th Annual Conference on Neural Information Processing Systems (NIPS). (2010) 1153–1161
8. Parson, O., Ghosh, S., Weal, M., Rogers, A.: Non-intrusive load monitoring using prior models of general appliance types. In: Proceedings of Twenty-Sixth Conference on Artificial Intelligence (AAAI-12), Toronto, CA (July 2012)
9. Gatsis, N., Giannakis, G.B.: Residential load control: Distributed scheduling and convergence with lost ami messages. Smart Grid, IEEE Transactions on **PP**(99) (2012) 1 –17
10. Mohsenian-Rad, A.H., Leon-Garcia, A.: Optimal residential load control with price prediction in real-time electricity pricing environments. Smart Grid, IEEE Transactions on **1**(2) (sept. 2010) 120 –133
11. Tischer, H., Verbic, G.: Towards a smart home energy management system - a dynamic programming approach. In: Innovative Smart Grid Technologies Asia (ISGT), 2011 IEEE PES. (nov. 2011) 1 –7
12. Yu, Z., McLaughlin, L., Jia, L., Murphy-Hoye, M.C., Pratt, A., Tong, L.: Modeling and stochastic control for home energy management. In: 2012 Power and Energy Society general meeting. (July 2012)
13. Kim, T., Poor, H.: Scheduling power consumption with price uncertainty. Smart Grid, IEEE Transactions on **2**(3) (sept. 2011) 519 –527
14. Levorato, M., Goldsmith, A., Mitra, U.: Residential demand response using reinforcement learning. In: in Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on. (2010) 409–414
15. Chen, Z., Wu, L., Fu, Y.: Real-time price-based demand response management for residential appliances via stochastic optimization and robust optimization. Smart Grid, IEEE Transactions on **3**(4) (dec. 2012) 1822 –1831