AgentSwitch: Towards Smart Energy Tariff Selection

Sarvapali D. Ramchurn,[1] Michael A. Osborne,[2] Oliver Parson,[1] Talal Rahwan,[1] Sasan Maleki,[1] Steve Reece,[2] Trung D. Huynh,[1] Mudasser Alam,[1] Joel E. Fischer,[3] Tom Rodden,[3] Luc Moreau,[1] Stephen Roberts[2]

[1]Electronics and Computer Science, University of Southampton, United Kingdom
{sdr,op106,tr,sm1g09,tdh,ma606,lavm}@ecs.soton.ac.uk

[2]Pattern Recognition Lab, University of Oxford, United Kingdom
{mosb,sjrob}@oxford.ac.uk

[3]Mixed Reality Lab, University of Nottingham, United Kingdom
{Joel.Fischer,Tom.Rodden}@nottingham.ac.uk

ABSTRACT

In this paper, we present AgentSwitch, a prototype agent-based platform to solve the electricity tariff selection problem. AgentSwitch incorporates novel algorithms to make predictions of hourly energy usage as well as detect (and suggest to the user) deferrable loads that could be shifted to off-peak times to maximise savings. To take advantage of group discounts from energy retailers, we develop a new scalable collective energy purchasing mechanism, based on the Shapley value, that ensures individual members of a collective (interacting through AgentSwitch) fairly share the discounts. To demonstrate the effectiveness of our algorithms we empirically evaluate them individually on real-world data (with up to 3000 homes in the UK) and show that they outperform the state of the art in their domains. Finally, to ensure individual components are accountable in providing recommendations, we provide a novel provenance-tracking service to record the flow of data in the system, and therefore provide users with a means of checking the provenance of suggestions from AgentSwitch and assess their reliability.

Categories and Subject Descriptors

I.2.11 [**Computing Methodologies**]: Artificial Intelligence—*Distributed Artificial Intelligence*

Keywords

Electricity, Smart Grid, Optimisation, Group Buying, Provenance, Recommender Systems

1. INTRODUCTION

Energy poverty is a rapidly growing issue across the world due to the significant rise in energy costs over the last few years.[1] Such increments are due to the unprecedented growth in energy demand (e.g., world energy is set to grow by more than 50% by 2030) coupled with dwindling fossil fuels and the high costs of (and resentment against) constructing large renewable energy generation facilities. This is set against a background where ageing nuclear and coal power stations are gradually being decommissioned (and as a result, a recent report suggested widespread blackouts are to be expected in the United Kingdom by 2015 [10]). In the UK energy market, we note that the loss of generation capacity is exacerbated by a misalignment of incentives whereby energy companies have no motivation to innovate nor to adopt cleaner sources of energy, given that they can easily pass on any rising costs they face directly to their customers. Thus, unsurprisingly, retailers have made significant profits even during periods of serious economic downturn[2]. In addition, most consumers (40%-60% in the UK) tend to 'stick' to the same energy supplier year in year out and do not spend much time looking for a cheaper deal. This reduces competition in the energy retail market and does not help drive down prices [11, 19].[3]

Now, consumers cannot be completely blamed for not finding the best deal, given that energy tariffs are often made explicitly complex and, at times, confusing. For example, tariffs may have multiple tiers (e.g., the first tier may be priced at 20p per kWh, and beyond this the cost drops to 5p/kWh), implement time-of-use pricing (e.g., 5p/kWh between 11pm and 7am), and may include additional one-off discounts (often only for a limited period). To help consumers combat this complexity, a number of third-party online services exist to help consumers submit simple estimates of their yearly consumption and obtain the cheapest tariff (e.g., uswitch.com and moneysupermarket.com). Some other services also claim to help consumers come together as a collective in order to access group discounts from retailers (e.g., which.com and incahoot.com). Crucially, however, these services rely on consumers being able to make a reasonable estimate of their yearly consumption (taking into account varying usage over different seasons and usage at on- and off-peak times) and being able to understand how to take advantage of the various tiers or time-of-use tariffs they offer (e.g., by shifting appliance usage to off-peak times). Moreover, in existing collective purchasing systems, all members of the collective tend to obtain the same contract without

[1]According to the British energy regulator, *ofgem*, half a million households in the UK have been put into fuel poverty due to price increases in 2008 alone, while domestic energy bills have nearly doubled since 2003.

[2]One of the largest energy suppliers in the UK noted a rise in profits of more than 20% in the first half of 2012 despite the UK economy being in recession.

[3]Indeed, research by the U.S. Dept. of Energy found that most people are likely to spend *no more than two hours a year* setting their preferences for comfort, tariffs, and environmental impact [19].

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013), Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.*

considering whether the discounts are fairly distributed across all the members of the collective (e.g., those with unpredictably peaky consumption profiles should be charged more than those with predictably flat profiles, as they tend to cause higher penalties in the balancing market).

Against this background, here we report on the development of a prototype agent-based platform, called AgentSwitch[4], that integrates state-of-the-art techniques and mechanisms to address the challenging issue of energy tariff selection. AgentSwitch builds upon the data provided by off-the-shelf energy monitoring devices, and applies a number of machine learning, optimisation, and coalition formation algorithms in order to solve the energy tariff selection problem. In more detail, this work advances the state of the art in the following ways. First, we develop novel extensions to Bayesian Quadrature (a machine learning technique), in order to generate predictions of yearly consumption at hourly level. These estimates can then be directly used to select the best tariff available from energy retailers. Second, using the predictions of yearly consumption, we develop a novel mechanism for collective energy purchasing. The mechanism relies on a novel scalable algorithm to approximate (in linear time) the Shapley value for coalitional games involving thousands of agents (homes). Third, we present a novel non-intrusive appliance load monitoring (NIALM) algorithm that works on coarse energy data (at five-minute level rather than second-level as is traditionally the case in this field) in order to detect deferrable loads that might benefit from being shifted to off-peak times. This algorithm is shown to outperform the state of the art on standard datasets. Fourth, we implemented a novel provenance service that allows the tracking of data throughout the system in order to provide accountability for its recommendations. As such, AgentSwitch instantiates the foundational tools, substantiated by benchmarks against the state of the art, in order to address a real world challenge for real users.

The rest of this paper is structured as follows. Section 2 details the system architecture underlying AgentSwitch. Section 3 elaborates on our Bayesian Quadrature model to predict yearly energy consumption from limited data. Section 4 then details our group buying mechanism that includes techniques to form coalitions via clustering of homes with similar attributes, along with a novel scalable Shapley value computation algorithm that allows for approximately fair distribution of energy costs across consumers. Section 5 then presents our load disaggregation algorithm that helps provide better suggestions to users. Section 6 presents our provenance tracking mechanism. Finally, Section 7 concludes the paper.

2. AGENTSWITCH ARCHITECTURE

AgentSwitch is implemented as a prototype web application that incorporates a number of key modules (see Figure 1) as follows: (i) an annual load prediction module that uses Bayesian Quadrature (BQ) to predict annual electricity consumption, (ii) a group buying module that identifies and forms coalitions of consumers to take advantage of group discounts from retailers, (iii) a load disaggregation module that uses NIALM techniques to analyse home-level electricity readings to identify opportunities for savings by shifting appliances to off-peak times, (iv) a provenance tracking service that tracks the flow of data in the system, and finally (v) user interfaces to input data and visualise recommendations in such a way that the complexity of the processes underlying AgentSwitch are hidden from view.

Now, in order to provide tariff recommendations to users who sign up to use the service (i.e., allow AgentSwitch to analyse their data to provide recommendations and formulate group buying strategies), AgentSwitch needs to access at least[5] two key sources, namely consumers' electricity consumption readings (to be kept in a database, termed Energy DB, in AgentSwitch for algorithms to analyse) and live electricity tariff specifications from all retailers (for AgentSwitch to match consumption predictions or group consumption against the best tariffs).[6] While the former can be obtained from users' off-the-shelf energy monitors or smart meters that provide average power readings at different levels of granularity, the latter can be obtained from online third-party providers and suppliers (in our case, we used live tariff data from `uswitch.com` with their permission).

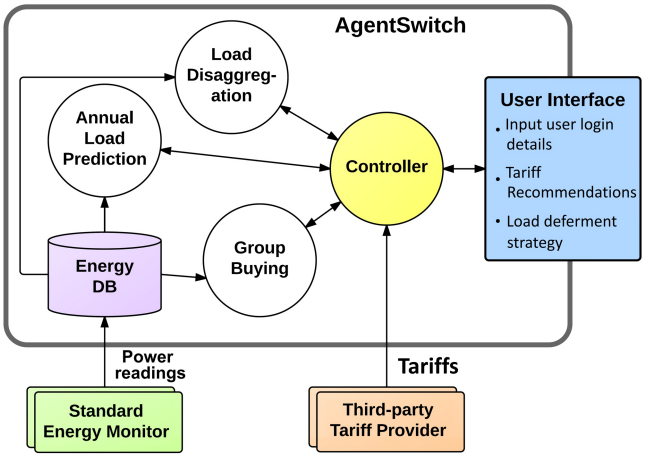

Figure 1: The architecture of AgentSwitch showing the data flows (arrows) between different modules. The circles represent algorithms, rectangles represent data providers or sinks. All the data flows in the system are tracked by our provenance service.

In what follows, we individually detail the core computational components of AgentSwitch as presented above. While we omit detailed descriptions of the architectural elements enabling the efficient storage and analysis of data in the Energy DB as well as user interface elements, they constitute important features of AgentSwitch that render it fit for a real-world deployment. Thus, we choose to focus on the core Artificial Intelligence (AI) elements of AgentSwitch in what follows.[7]

3. ANNUAL LOAD PREDICTION

The choice of an electricity tariff is based on a consumer's annual energy consumption and, to date, most online services would normally use a consumer's monthly usage and correlate this with the national average (for a given type of consumer, e.g., a large family in a large house, or a single living in a flat) in order to arrive

[4]see `http://agentswitch.orchid.ac.uk`.

[5]Other sources of data such as weather predictions and historical data, users' schedule and travel plans for the year ahead, the users' home types and contents, would all be useful to improve predictions and load disaggregation and could be integrated into our approach. Thus, despite its complexity, AgentSwitch in its current shape represents only the first step towards building fully integrated tariff selection and home energy management systems.

[6]As a first step, we only consider electricity due to the wide availability of such electricity monitors, but this approach could be easily extended to gas or water if the appropriate sensors become available and affordable.

[7]As this is more relevant to the Agents community.

at an estimated annual consumption (possibly ignoring the division between peak and off-peak consumption). In contrast, here, we present a principled approach to computing such estimates in order to select the best tariff for a given customer. To this end, we employ a Gaussian process (GP) to model power consumption as a function of time. In particular, we apply this Gaussian process model to estimate the integral of power consumption over a year (the total annual energy consumption) so that estimates of time-of-use costs are correctly computed. This technique is known as Bayesian Quadrature (BQ) [12, 4, 15], a model-based means of numerical integration. To date, only very simple GP covariance functions (typically Gaussian) have been used to perform BQ. In this work, we extend Bayesian Quadrature techniques to use a more sophisticated GP covariance, suitable for quasi-periodic signals. Such signals emerge from the weekly cycles of energy consumption that typical domestic consumers exhibit.

3.1 Bayesian Quadrature

Bayesian quadrature [12, 15] is a means of performing Bayesian inference about the value of a potentially nonanalytic integral,

$$\langle f \rangle := \int f(x)p(x)\mathrm{d}x. \tag{1}$$

For clarity, we henceforth assume the domain of integration $\mathcal{X} = \mathbb{R}$, although all results generalise to $\mathbb{R}^n$. Previous work on BQ assumes a Gaussian density $p(t) := \mathcal{N}(t; \nu_t, \lambda_t)$, although other convenient forms, or, if necessary, the use of an importance reweighting trick ($q(x) = {}^{q(x)}/_{p(x)} p(x)$ for any $q(x)$), allow any other integral to be approximated.

Quadrature involves evaluating $f(t)$ at a vector of sample points $\mathbf{t}_s$, giving $\boldsymbol{f}_s := f(\mathbf{t}_s)$. Often this evaluation is computationally expensive; the consequent sparsity of samples introduces uncertainty about the function f between them, and hence uncertainty about the integral $\langle f \rangle$.

Previous work on BQ chooses a Gaussian process (GP) [16] prior for f, with mean μ_f and squared exponential (or un-normalised Gaussian) covariance function, suitable for modelling very smooth functions,

$$K_{SE}(t_1, t_2|h,\, w) := h^2 \, \exp -\frac{1}{2}\left(\frac{t_1 - t_2}{w}\right)^2 \tag{2}$$

Here hyperparameter h specifies the output scale over f, while hyperparameter w defines an input scale over t. This covariance can be readily modified to give a periodic covariance, suitable for modelling periodic functions,

$$K_{PSE}(t_1, t_2|h,\, w,\, P) := h^2 \, \exp -\frac{1}{2}\left(\frac{1}{w}\sin\left(\pi\frac{t_1 - t_2}{P}\right)\right)^2 \tag{3}$$

where h and w are hyperparameters that have interpretations as for the squared exponential, and P is the hyperparameter representing the period.

We use the following dense notation for the standard GP expressions for the posterior mean m and covariance Σ, respectively: $m_{f|s}(t_\star) := m(f_\star|\boldsymbol{f}_s)$ and $\Sigma_{f|s}(t_\star, t'_\star) := \Sigma(f_\star, f'_\star|\boldsymbol{f}_s)$.

Variables possessing a multivariate Gaussian distribution are jointly Gaussian distributed with any affine transformations of those variables. Because integration is affine, we can hence use computed samples $\boldsymbol{f}_s$ to perform analytic Gaussian process inference about the value of integrals over $f(t)$, such as $\langle f \rangle$. The mean estimate for $\langle f \rangle$ given $\boldsymbol{f}_s$ is

$$\begin{aligned} m(\langle f \rangle|\boldsymbol{f}_s) &= \iint \langle f \rangle \, p(\langle f \rangle|f) \, p(f|\boldsymbol{f}_s) \, \mathrm{d}\langle f \rangle \, \mathrm{d}f \\ &= \iint \langle f \rangle \, \delta\Big(\langle f \rangle - \int f(t)\, p(t)\, \mathrm{d}t\Big) \, \mathcal{N}\big(f; m_{f|s}, \Sigma_{f|s}\big) \, \mathrm{d}\langle f \rangle \, \mathrm{d}f \\ &= \int m_{f|s}(t)\, p(t)\, \mathrm{d}t\,, \end{aligned} \tag{4}$$

which, for Gaussian input density and squared exponential covariance, is expressible in closed-form due to standard Gaussian identities [15].

3.2 Applying BQ to Household Energy Consumption Prediction

We applied our model to predict the yearly energy consumption of a test set of UK homes. This application motivated the development of a bespoke GP mean and covariance to permit long-range forecasting. The prior mean function was taken as:

$$\mu(t|a) := c\,\alpha(t), \tag{5}$$

where c is a hyperparameter acting as a scaling constant, and $\alpha(t)$ is the UK national average energy consumption for time-of-use consumers drawn from data provided by Elexon Ltd. Figure 2 illustrates this national average consumption over the year. Our motivation for this prior mean was to permit the model to appropriately extrapolate from the patterns of consumption in winter to the remaining seasons in 2012. Then, the covariance chosen was built from terms of the form (2) and (3),

$$\begin{aligned} &K(t_1, t_2|h_a, w_a, w_p, P, h_b, w_b) \\ &:= K_{SE}(t_1, t_2|h_a, w_a) K_{PSE}(t_1, t_2|h = 1, w_p, P) \\ &\qquad + K_{SE}(t_1, t_2|h_b, w_b), \end{aligned} \tag{6}$$

modelling the quasi-periodic behaviour of the signal (e.g., lower energy use at night). Note that this covariance introduces additional hyperparameters. The period P is set *a-priori* to one day; the daily period was by far the most significant repeated pattern in the data. Other hyperparameters (including an additional observation-noise variance hyperparameter) are fitted using maximum likelihood for each available data-set.[8] Figure 3 displays an example of GP regression using the model above.

We ultimately wish to compute an integral of power over time in order to estimate energy computation. That is, we need to solve (1) for an input density $p(t)$ that is constant for the desired time period and zero elsewhere. Unfortunately, the complex form of (6) rules out the analytic computation of (4) required for traditional BQ. In order to effect BQ for our problem, we discretise by assuming power is constant for each observed minute, and aggregate to provide observations of total energy consumption in each observed hour. We can then use these within the BQ framework as noisy integral observations of the true power signal. These observations can then be used to directly infer the desired integral over the entire year. The use of the BQ formalism allows for both a mean and variance to be provided for such integrals. Thus, the flexibility of our model allows for inference of the integral of power over any desired time period. Use of the time-of-use tariffs implies different electricity prices in nominal night and day periods, in order to encourage electricity demand at night. As such, in order to provide recommendations to users of the benefits of switching to or

[8] Note that we do not yet consider correlations between houses. Future work might investigate the emergence of such correlations from observable demographic information, such as household income and the number of household residents.

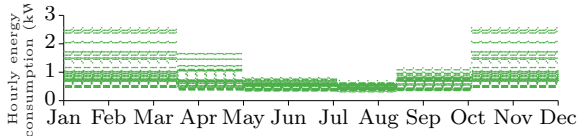

Figure 2: UK national average hourly energy consumption throughout the year.

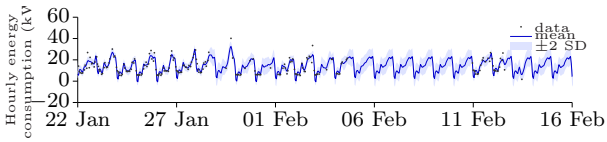

Figure 3: Example of GP regression for energy consumption given sparse observations.

from tariffs with cheaper electricity during the night (such as the UK's *Economy 7* tariff), we must estimate both annual day-time and night-time energy consumption. This is trivially achieved by our approach.

3.3 Evaluating BQ

We evaluated the efficacy of our prediction module on half-hourly consumption data collected for a set of 18 UK homes over winter 2011-2012 (our longest metered set of homes out of a pool of 3000) and divided the data into a set of training data (one third of the total data) and test data (the remaining two-thirds); the goal was to accurately predict the total energy usage across the test period given only training data. To provide comparisons for the BQ method, we implemented two alternative methods. The first predicted that energy usage for the household would be identical to the national average, ignoring the data. The second assumed that energy usage for the test data would continue at exactly the mean usage rate across the training data; this method ignores prior knowledge about the national pattern of energy usage. The root mean square errors (RMSEs) across all households are tabulated in Table 1. These experiments reveal that in combining both prior knowledge and the information contained in data, BQ returns slightly better predictions than either alternative. While the improvement offered by BQ for this limited segment of data are modest, we are hopeful that datasets spanning entire annual cycles (or taking into account localised weather patterns) will permit the full modelling capacity of BQ to achieve more significant gains. We expect that explicitly encoding a non-negativity constraint [13] for energy usage may also improve the BQ model. Note that in addition to providing mean estimates of energy consumed, the probabilistic BQ method also provides *an indication of the uncertainty in such quantities* in terms of the variance in yearly consumption. The provision of such uncertainty estimates is crucial to helping consumers understand the degree of confidence that can be attributed to the savings that a given tariff could provide. Thus, if a consumer has a large consumption variance, he or she may prefer to choose a tariff which is less risky (e.g., fixed average price rather than one that charges extra for going above a given tier).

Table 1: RMSE **(MWh) for predictions of total energy usage.**

Nat. Avg.	Mean	BQ
4.57	0.28	**0.27**

By predicting consumption accurately using the techniques above, we are able to choose the best tariff for any given home (resulting in savings of hundreds of pounds depending on the previous tariff). In the next section, we extend the use of such predictions to help create efficient energy purchasing collectives in order to take advantage of group discounts.

4. COLLECTIVE ENERGY PURCHASING

The creation of consumer collectives is a challenging problem given the number of potential consumers that can be grouped, along with the vast amounts of historical energy consumption data they come with. To date, apart from generic signup mechanisms, there is no principled way to group energy consumers and take advantage of group discounts (e.g., forming groups with flat energy profiles that enable the purchase of energy contracts in the forward market or green groups that prefer power from renewable supplies). A key challenge is the problem of sharing the group discounts "fairly" among group members. In this section, we describe a principled approach to do so based on the concept of the *Shapley value* from cooperative game theory [18], and address the computational issues that arise by presenting a novel, scalable algorithm that is integrated within AgentSwitch, and is shown to significantly outperform the state of the art.

In what follows, we first describe how we model the collective energy purchasing problem as a cooperative game involving coalitions (i.e., non-overlapping groups) of consumers, and discuss the challenges involved in computing the Shapley value. Second, we describe our algorithm for computing the Shapley value. Finally, we evaluate our algorithm using real data.

4.1 Cooperative Game Model

Given a group of consumers and their estimates, the *value* of a coalition (i.e., a group of energy consumers signed up to AgentSwitch) is the expected cost of the aggregated consumption of all its members, taking into consideration the group discount. Due to the workings of the forward energy market, and the penalties applied to unexpected over-consumption and under-consumption in the balancing market, the consumers that will obtain the highest discounts are those whose consumption estimate has a low variance. In this vein, adopting a similar model to [17, 1], we construct the expected cost function for a coalition C based on the expected usage profile over a given period (e.g., day, month, or year), as well as the variances (which captures the uncertainty in demand) of all coalion members. More formally, let A be the set of consumers with an AgentSwitch account, and $C \subseteq A$ be a potential coalition. Moreover, let $f_i(S) \sim \mathbb{N}(m_i(S), \Sigma_i(S,S))$ be the predicted consumption profile for consumer $i \in C$ over times S. Now, for coalition C, the mean predicted consumption is:

$$\mu_C = \sum_{i \in C} \int_S m_i(t) dt \; .$$

Furthermore, the variance of the predicted consumption is:

$$V_C = \sum_{i \in C} \int_S \int_S \Sigma_i(t, t') \; dt \; dt'$$

Based on these, the characteristic function, which is a function that reflects a coalition's expected cost, is:

$$v(C) = p \times \mu_C - k\sqrt{V_C} \tag{7}$$

where p is the unit price, and k is a constant parameter that captures the balancing penalties. We can replace the integals above with summations for discrete times S. Given the above cost function for

a coalition of homes, it turns out that the minimum cost is achieved if all consumers come together in the *grand coalition* (i.e., the set of all consumers). With this in mind, the objective of AgentSwitch is to divide the value of the grand coalition, i.e., the expected cost, among its members in a fair way.

We start by noting that the presence of any consumer i in a coalition C causes a reduction in the expected cost that the coalition has to pay. This reduction is referred to as the *marginal contribution* of consumer i to coalition C, and is given by $v(C \cup \{i\}) - v(C)$. According to the Shapley value, the fairest way to divide the group discount is as follows. Each consumer receives a share that is equal to his average marginal contribution to all possible subsets of the rest of the consumers. This division satisfies a number of axioms, each of which being a desirable fairness property.

Unfortunately, due to its combinatorial nature, the Shapley value quickly becomes hard to compute as the size of the grand coalition exceeds tens. The state-of-the-art research on addressing this issue [2] proposes an approximation algorithm based on uniform sampling from the marginal contributions. However, since the approximation bound that this technique provides may fail with some probability, it is undesirable to use it for dividing consumers' payments, as it might negatively impact their willingness to sign up to AgentSwitch. Moreover, depending on the variance and the desired probability of failure, the number of samples needed to be used could actually exceed the total number of marginal contributions.

Against this background, in the next subsection, we present a branch-and-bound algorithm for computing the Shapley value of a given consumer. Initially, this algorithm approximates the Shapley value, and establishes a tight bound, both in linear time. After that, it iteratively improves the approximation until it eventually arrives at the exact value.

4.2 Computing the Shapley Value for Group Discounts

For any consumer, $i \in C$, let $MC^i \subseteq \mathbb{R}$ be the set of marginal contributions of i (to all possible coalitions). As mentioned earlier in Section 4.1, the Shapley value of i is the average of MC^i. The challenge comes from the exponentially large number of elements in MC^i ($2^{|C|-1}$ to be precise). To overcome this challenge, we divide MC^i into $n-1$ subsets (where n is the number of agents), namely: $MC^i_s : s = 0, \ldots, n-1$, where each MC^i_s consists of the marginal contributions of i to coalitions of size s. The rationale behind this division is that it results in a number of subsets linear in the number of consumers. We note that, with some characteristic functions, such as the one we introduced in equation (7), the maximal and minimal elements of each MC^i_s can be found in constant time. Based on this observation, we approximate the average of each MC^i_s as the mid-range of its maximal and minimal elements. This way, the worst-case error (i.e., the distance between the approximated and the actual value) can also be quantified. Setting the complexity of computing $v(C)$ aside, this approximation requires a number of operations linear in the number of consumers.

Once the algorithm has computed the initial approximation, and has established the tight bound, it iteratively improves both the approximation and the bound until it reaches the actual Shapley value of the consumer. This is done as follows. In each iteration, the algorithm selects what we call a *branching agent*, and divides MC^i_s into two subsets: those elements whose corresponding coalitions *contain* the branching agent, and those whose corresponding coalitions *do not contain* it. Then, for each one of these two subsets, the algorithm finds the maximal and minimal elements, and uses them to approximate the average value in that subset in the same way that it would initially do with MC^i. If more time is available, the algorithm divides each one of the two parts of MC^i_s further based on yet another branching agent, and so on and so forth. By continuing this process, the algorithm reaches a state where MC^i has been divided into 2^{n-1} subsets. In this case, the approximated average of each subset would be equal to its actual average.

Algorithm	No. MC	Avg. Error %	Confidence
AgentSwitch	1200	0.4%	100%
Castro [2]	2950	0.4%	75%
Liben-Nowell [7]	25×15000	0.4%	75%
Optimal	25×2^{24}	0	100%

Table 2: Comparison of our algorithm and the state-of-the-art, for 25 randomly chosen consumers from a pool of 3000 UK homes.

Table 2 shows a comparison of the performance of our algorithm, and two state-of-the-art algorithms. As the expected cost function mentioned above happens to be *supermodular*, one of the two algorithms that we benchmark against is the polynomial sampling algorithm proposed by Liben-Nowell et al. [7] for supermodular games. For the sake of also including the exact Shapley value, which has an exponential time complexity, we have considered 25 randomly chosen consumers from a pool of 3000 UK homes. As can be seen, the total number of marginal contributions that our algorithm requires to approximate the Shapley value of all the 25 consumers with an average error of 0.4% is a significantly small number compared to the 25×2^{24} optimal. The next best approximation is Castro et al.'s [2], which requires to evaluate 145% more marginal contributions than our algorithm to approximate with the same precision. Whereas our algorithm does so with 100% confidence on the reported error, Castro et al.'s and Liben-Nowell et al.'s confidences are 75%, which can only be improved with more samples (i.e., more computation).

In separate tests, given the linear complexity of our algorithm and considering 3000 homes in the grand coaltion, our algorithm took an average of 4 seconds to compute the Shapley value of a home. Given these values, it is then possible to specify a tariff for every customer, along with possible penalties for deviating from the expected consumption.

We next address the issue of making the most of a given (time-of-use, real-time, or group) tariff by finding out which loads can be deferred to off-peak times given only an aggregate measure of power readings from homes.

5. APPLIANCE DISAGGREGATION

Previous work has shown that household electricity data can be disaggregated into individual appliances, therefore enabling suggestions for optimising a household's energy efficiency to be provided to a household's occupants [14]. The aim of such feedback is to provide well-defined actionable suggestions (e.g., shift your five washing machine loads or two dishwasher loads to off-peak times) with clear savings rather than leave it to consumers to decide which appliances they should shift to make any significant savings [3]. However, disaggregating appliances from a household's electricity consumption in the context of AgentSwitch is a challenging problem for the following three reasons. First, the data sampling rate of once every 5 minutes (as obtained from off-the-shelf meters) is less frequent than that required by similar hidden Markov model (HMM) based approaches [5, 14]. Second, the data available to AgentSwitch represents the average power demand over the

sampling interval, as opposed to the instantaneous power demand, therefore blurring the changes in the observed power demand when an appliance turns on or off. Third, the number and type of appliances within each household are not known. As a result, the methods proposed in previous literature are not applicable, and instead a new approach is required.

Thus, AgentSwitch focuses on the identification of appliances which both consume a large amount of energy and can be deferred to another time of day with minimal inconvenience to the household occupants. Three common examples of such deferrable appliances are the washing machine, clothes dryer and dishwasher, and we shall refer to the use of such appliances as *deferrable loads*. Since each deferrable load has a high energy consumption, they have the advantage that they can be disaggregated from the remainder of a household's energy consumption, despite the low data sampling rate. Given this, we first construct appliance models from individually metered appliances from houses other than those in which disaggregation will be performed, which we refer to as the training phase. Second, the appliance models are used to identify appliance signatures within the aggregate electricity data, which we refer to as the disaggregation phase. The training phase consists of operation detection, followed by feature extraction and model construction, while the disaggregation phase consists of operation detection, followed by feature extraction and operation classification (i.e., identifying the appliance run). The following sections describe each of these processes.

5.1 Operation Detection

The aim of operation detection is to identify pairs of switch events from raw power data, potentially corresponding to an appliance turning on, followed by the same appliance turning off. A switch event, e_n, is defined by an increase or decrease in the raw power, $\mathbf{p}$, within a range of values defined by the appliance model: $p_{min} < |p_t - p_{t-1}| < p_{max}$.

Possible appliance operations are then identified as positive switch events, e_{start}, followed by a negative switch event, e_{end}, separated by a duration within a range defined by the appliance model: $d_{min} < t_{e_{end}} - t_{e_{start}} < d_{max}$.

5.2 Feature Extraction

A number of features are then extracted from the detected operations. In the training phase, these features are used to build the appliance models, while in the disaggregation phase, these features are used to determine how well a detected operation matches an appliance model. We extract the following seven features from each operation: on power, off power, duration, power range, ratio of high to low power readings, minimum energy and number of peaks in power demand. We refer to these features as $x_1, \ldots, x_7$, respectively.

5.3 Model Construction

In the training phase, we construct models of each deferrable appliance by fitting known distributions to the output of the feature extraction module by maximum likelihood. We use a two-dimensional Gaussian probability density function (PDF) for the on and off power, since these features are interdependent, and we use a single-dimensional Gaussian PDF for the appliance duration, power range and ratio of high to low power readings. In addition, we use a single-dimensional Gaussian cumulative density function (CDF) for the minimum energy feature, to provide a smooth lower bound. Last, we use a switch function to match the number of peaks in the power demand. We denote the parameters of these six functions as $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_6\}$, respectively. In addition, we also extract boundary values for the appliances' power, p_{min} and p_{max}, and the appliances' duration, d_{min} and d_{max}.

5.4 Operation Classification

In the disaggregation phase, to determine whether a potential appliance operation is a deferrable load, we define a likelihood function which describes the similarity between a potential appliance operation and the previously learned appliance model. The likelihood function is defined by:

$$\begin{aligned}\mathcal{L}(e_{start}, e_{end}; \boldsymbol{\theta}) =& f(x_1, x_2; \theta_1) f(x_3; \theta_2) f(x_4; \theta_3) \\ & f(x_5; \theta_4) F(x_6; \theta_5) g(x_7; \theta_6)\end{aligned} \tag{8}$$

where the function f represents the PDF of a Gaussian distribution parameterised by $\boldsymbol{\theta}$, the function F represents the CDF of a Gaussian distribution parameterised by $\boldsymbol{\theta}$ and g represents a function which returns 1 if x_7 is equal to the number of peaks in the model and 0 otherwise.

The likelihood of an operation is compared to a likelihood threshold, L, to determine its classification as a deferrable load: $\mathcal{L}(e_{start}, e_{end}; \boldsymbol{\theta}) > L$. The likelihood threshold was optimised using submetered training data from houses other than those in which the disaggregation is being performed.

5.5 Evaluating Load Disaggregation

In order to test the accuracy of the AgentSwitch disaggregation algorithm, we required a data set for which each appliance's power demand is known, since the homes we monitored (as detailed in Sections 3 and 4) did not have appliance level monitoring and hence cannot be used for this test. Instead, the Reference Energy Disaggregation Dataset (REDD) [6] is such a data set, in which both the household aggregate and individual appliance power demands were monitored. We selected three houses which contained a dishwasher, and downsampled the data to 5 minute average power readings to mimic the UK homes we monitored.

We benchmarked the AgentSwitch approach against a state-of-the-art HMM-based approach [14]. Such approaches predominantly use step changes in power demand and state durations as their primary features. In addition, they represent appliances using state transition models, which describe the probability of a transition from one state (e.g., on) to another state (e.g., off) given the observed power readings.

We compare the accuracy of the AgentSwitch and HMM-based approaches using three metrics: precision, recall and F-score. These metrics respectively represent the fraction of detections which were actually loads, the fraction of loads which were detected and a weighted average of the two.

The detection accuracy of the AgentSwitch and the HMM-based approaches using the REDD data set are shown in Table 3. It can be seen that the AgentSwitch approach outperforms the HMM-based approach. This is due to the nature of the low-granularity average power readings, which mimics the granularity of UK smart meter data. Consequently, instantaneous increases in the power demand caused by an appliance turning on or off are blurred across consecutive readings due to the averaging. The HMM-based approach is not robust to such blurring, and consequently suffers a loss in accuracy. Conversely, the AgentSwitch approach utilises additional features even when the on and off event signatures are blurred, and as a result is robust to such effects. So far, we have considered homes in isolation and how to provide feedback to consumers in order to help them maximise their savings given a time-of-use electricity tariff. In our test homes, the savings from deferring the identified loads were not major (limited to less than £50 a year) due to low usage of deferrable appliances at peak times electricity. For higher us-

Table 3: Accuracy metrics of dishwasher detection

Approach	Precision	Recall	F-score
AgentSwitch	0.687	0.600	0.596
HMM	0.568	0.438	0.456

age levels at peak times (e.g., large families or flats) and as energy prices rise and more complex time-of-use pricing (e.g., real-time or critical peak pricing) are put in place to combat peaks in demand, we expect such savings to increase substantially. A key step, however, to achieve these savings is to ensure consumers can understand and trust the recommendations provided by AgentSwitch. To support mechanisms and interfaces to do so, we next introduce our provenance-tracking service.

6. PROVENANCE IN AGENTSWITCH

In order to ensure consumers understand and build confidence in the suggestions given by AgentSwitch, it is important to provide them with a means to track the flow of data and decisions made throughout the system. For example, using such traces, users (or other services acting on their behalf) may be able to identify faulty sensors, incorrect assumptions about yearly consumption, or justify changes to daily routine to make significant savings.

As shown in the previous sections (see Figure 1) various types of data come in/out of AgentSwitch modules and get consumed or transformed at the same time (e.g., electricity consumption data from a third-party provider, post code from user input, tariffs from uswitch.com). Therefore, the quality of recommendations given by AgentSwitch depends on the quality of data it receives from the different sources and the performance of its individual components. As recommendations from AgentSwitch might potentially lead to financial gains or losses, it is crucial to be able to identify the origin of error once a recommendation is deemed to be inaccurate. This, however, is challenging due to the multiple paths of (data) dependencies inherent to such a complex system.

In order to address this issue, the chains of dependencies that lead to a recommendation, i.e., its provenance,[9], were fully tracked in AgentSwitch. Such information enables a systematic approach to pinpointing the sources or agents responsible for the errors and to auditing the data produced within the system. In more detail, the integration of provenance tracking in AgentSwitch is described in Section 6.1. Section 6.2 then demonstrates various use cases of provenance in the application.

6.1 Tracking provenance

Provenance in AgentSwitch was modelled using the PROV Data Model (PROV-DM) [9] being standardised at the World Wide Web Consortium. In this model, the three main types of element are: entity — a physical or digital thing (e.g., a piece of data), activity — something that occurs and generates or changes entities, and agent — something that bears some form of responsibility for an activity. In addition to these elements, various kinds of relationships between entities, activities, and agents can be captured in PROV-DM (see [9] for more details). Figure 4, shows a visual representation of the provenance tracked in a disaggregation API call (to the load detection module described in Section 5). It shows that the API response `APIResponseDisaggregation_1` was derived from the disaggregation result (`DisaggregationResult_1`),

[9]Provenance is defined as a record that describes the people, institutions, entities, and activities involved in producing, influencing, or delivering a piece of data [9].

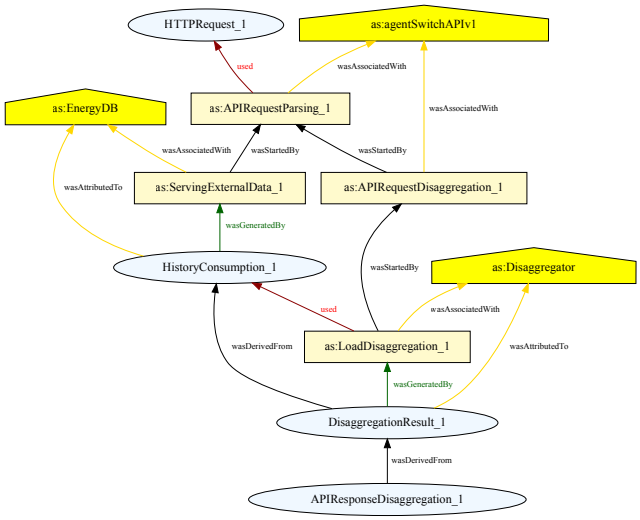

Figure 4: Provenance graph captured from a disaggregation API call to AgentSwitch. Boxes are activities that took place, pentagons the agents controlling them, and ovals the data entities consumed/produced.

which was attributed to a script in Matlab (`as:Disaggregator`). The activity `as:LoadDisaggregation_1`, in turn, used the electricity consumption data `HistoryConsumption_1` from an Energy DB (`as:EnergyDB`) to produce the disaggregation result.

Provenance tracking was integrated independently into each individual component of AgentSwitch. When a component passes its result to the next, the captured provenance also accompanies the data, such as from the AgentSwitch API to the web front-end. By so doing, the provenance graph for a recommendation evolves at each individual component, and the complete provenance is eventually made available to the web front-end for further use.

6.2 Exploiting provenance

Having had the provenance of activities in AgentSwitch captured, it was possible for the application to inform end users about its underlining processes in a number of ways:

- **Responsibility management**: In a similiar vein to [8], the responsibilities of contributing agents in a recommendation can be queried from its provenance. An example of which is presented in Figure 5, where parts of the load shifting advice are tagged with colour labels indicating the main modules responsible for the respective data. For instance, the AgentSwitch API (A) produced the cost calculation while the load disaggregation service (D) provided the appliance usage information in the screen-shot. We can also show that the costs were calculated from the selected tariff and the appliance usage information.[10]

- **Uncertainty management**: In addition to the dependencies between data entities, the provenance was tracked for metadata about those entities, such as the time span of consumption data, the uncertainty of annual consumption predictions (Section 3) and load disaggregation results (Section 5). Those uncertainty metrics can be propagated and combined in provenance graphs all the way to the final recommendation, allowing the estimation of its confidence level, which is then

[10]In this respect, a user interface for drilling down provenance graphs that offers an information representation suitable to end users is left as future work.

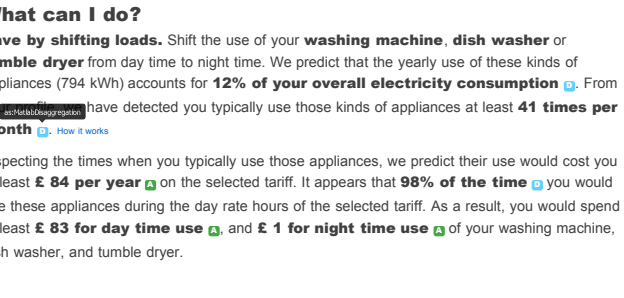

Figure 5: The screen-shot of a recommendation with colour tags (i.e. A and D) generated from tracked provenance indicating the agents that were mainly responsible for the various pieces of data shown.

presented to the user. This may help end users understand the risks associated with a recommendation and, as a result, adjust their decisions accordingly[11].

Beside the above benefits to end users, the captured provenance also showed us visually how AgentSwitch actually works, similar to a debug logging system to software developers. It allows the identification of the flow of executions in the system, the points where individual components interface with one another, as well as to manage data dependencies. In fact, provenance graphs produced by early versions of AgentSwitch had revealed inefficient code by showing duplicate (thus redundant) API calls, prompting us to improve the system.

7. CONCLUSIONS

In this paper we presented the core modules of AgentSwitch, an agent-based platform to enable smart energy tariff selection for domestic energy consumers. We focused on the key advances to the state of the art that the development of modules for AgentSwitch have brought about, including novel extensions to BQ, a novel algorithm for NIALM, and scalable algorithms to compute the Shapley value for energy purchasing collectives. Moreover, we showed how these modules can be packaged into an accountable information infrastructure underpinned by a novel provenance tracking service that allows users to track the flow of their data through individual modules and third-parties. While a working prototype of AgentSwitch, using live data from homes, has been completed[12], future work will look at integrating other sensor data (weather, consumer preferences) into the prediction and group buying algorithms in order to improve accuracy and the resulting savings.

Acknowledgments

This work is supported by the EPSRC-Funded ORCHID Project EP/I011587/1. We also thank uSwitch for access to their API.

8. REFERENCES

[1] E. Baeyens, E. Bitar, P. Khargonekar, and K. Poolla. Wind energy aggregation: A coalitional game approach. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 3000 –3007, dec. 2011.

[2] J. Castro, D. Gómez, and J. Tejada. Polynomial calculation of the shapley value based on sampling. *Comput. Oper. Res.*, 36(5):1726–1730, May 2009.

[3] J. Froehlich, L. Findlater, and J. Landay. The design of eco-feedback technology. In *Proc. CHI '10*, pages 1999–2008. ACM, 2010.

[4] M. Kennedy. Bayesian quadrature with non-normal approximating functions. *Statistics and Computing*, 8(4):365–375, 1998.

[5] J. Z. Kolter and T. Jaakkola. Approximate Inference in Additive Factorial HMMs with Application to Energy Disaggregation. In *International Conference on Artificial Intelligence and Statistics*, La Palma, Canary Islands, 2012.

[6] J. Z. Kolter and M. J. Johnson. REDD: A Public Data Set for Energy Disaggregation Research. In *ACM Special Interest Group on Knowledge Discovery and Data Mining, workshop on Data Mining Applications in Sustainability*, San Diego, CA, USA, 2011.

[7] D. Liben-Nowell, A. Sharp, T. Wexler, and K. Woods. Computing shapley value in supermodular coalitional games. In J. Gudmundsson, J. Mestre, and T. Viglas, editors, *Computing and Combinatorics*, volume 7434 of *Lecture Notes in Computer Science*, pages 568–579. Springer Berlin Heidelberg, 2012.

[8] S. Miles, S. Munroe, M. Luck, and L. Moreau. Modelling the provenance of data in autonomous systems. In *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'07)*, pages 1–8. ACM, 2007.

[9] L. Moreau and P. Missier. PROV-DM: The PROV Data Model. Candidate Recommendation, 2012.

[10] OFGEM. The retail market review - findings and initial proposals, 2011.

[11] OFGEM. Press release: projected tightening of electricity supplies reinforces the need for energy reforms to encourage investment., October 2012.

[12] A. O'Hagan. Bayes-Hermite quadrature. *Journal of Statistical Planning and Inference*, 29:245–260, 1991.

[13] M. A. Osborne, R. Garnett, S. J. Roberts, C. Hart, S. Aigrain, and N. Gibson. Bayesian quadrature for ratios. *Journal of Machine Learning Research - Proceedings Track*, 22:832–840, 2012.

[14] O. Parson, S. Ghosh, M. Weal, and A. Rogers. Non-intrusive Load Monitoring using Prior Models of General Appliance Types. In *26th AAAI Conference on Artificial Intelligence*, Toronto, Canada, 2012.

[15] C. E. Rasmussen and Z. Ghahramani. Bayesian Monte Carlo. In S. Becker and K. Obermayer, editors, *Advances in Neural Information Processing Systems*, volume 15. MIT Press, Cambridge, MA, 2003.

[16] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[17] H. Rose, A. Rogers, and E. H. Gerding. A scoring rule-based mechanism for aggregate demand prediction in the smart grid. In *The 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, June 2012.

[18] L. S. Shapley. A value for n-person games. *Contributions to the theory of games*, 2:307–317, 1953.

[19] US Department of Energy. The Smart Grid: An Introduction, 2007.

[11] Again, a provenance drilling-down interface will also be useful in this context, allowing end users to see where such uncertainty stemmed from. At the same time, it may help them gain an understanding of the inner workings of the system and potentially increase their trust in it.

[12] see `http://agentswitch.orchid.ac.uk`.