The Smart Appliance Scheduling Problem: A Bayesian Optimization Approach

Atena M. Tabakhi[1], William Yeoh[1], and Ferdinando Fioretto[2]

[1] Washington University in St. Louis, St. Louis, MO, USA
{amtabakhi,wyeoh}@wustl.edu
[2] Syracuse University, Syracuse, NY, USA
ffiorett@syr.edu

Abstract. Daily energy demand peaks induce high greenhouse gas emissions and are deleterious to the power grid operations. The autonomous and coordinated control of smart appliances in residential buildings represents an effective solution to reduce peak demands. This coordination problem is challenging as it involves, not only, scheduling devices to minimize energy peaks, but also to comply with user' preferences. Prior work assumed these preferences to be fully specified and known a priori, which is, however, unrealistic. To remedy this limitation, this paper introduces a Bayesian optimization approach for smart appliance scheduling when the users' satisfaction with a schedule must be elicited, and thus considered expensive to evaluate. The paper presents a set of ad-hoc energy-cost based acquisition functions to drive the Bayesian optimization problem to find schedules that maximize the user's satisfaction. The experimental results demonstrate the effectiveness of the proposed energy-cost based acquisition functions which improve the algorithm's performance up to 26%.

Keywords: Beyesian Optimization · Smart Appliance Scheduling Problem · User's Preference Elicitation · Constraint Satisfaction Problems

1 Introduction

Demand-side management (DSM) [14,19] in smart grids [7,6] consists of techniques and policies that allow customers to make informed decisions about their energy consumption while helping energy providers reducing peak load demand. DSM aims at encouraging customers to consume less energy during peak periods or shift their energy demand to off-peak periods [10]. To do so, a successful approach is to act on the schedule of shiftable loads, in which customers can shift the time of use for electrical appliances to off-peak hours [14,8].

Scheduling shiftable loads in residential buildings has become possible thanks to the advent of smart appliances (e.g., smart plugs, smart thermostats, smart washing machines, and smart dishwashers) [41]. These appliances enable users to automate their schedules and shift their executions during times when energy may cost less (e.g., under real-time electricity price [12,3]).

The *smart appliance scheduling* (SAS) problem in smart homes [25,8,41] is the problem of scheduling the operation of the appliances at any time of a day, subject to a set of user-defined constraints and preferences. Smart appliance scheduling aims at minimizing energy consumption and, consequently, reducing electricity costs [27]. Over the past few years, a large body of work has emerged to solve this problem. The majority of these works have focused on minimizing solely the energy consumption [3,42,1], ignoring users' *comfort/preferences* (see Section 2 for a thorough discussion). As a result, the solutions (i.e., appliances schedules) that these approaches provide may not be aligned with the users' preferences and, hence, may not be acceptable.

In response to this challenge, recent studies have formulated the appliance scheduling problem as a multi-objective optimization that minimizes users' discomfort and energy consumption [41,8,35]. To promote the users' comfort these studies assume perfect knowledge about the comfort information, which is not realistic in practice [2]. The work by Nguyen *et al.* contrasts this background and represents users' preferences as Normal distributions [20]. However, this approach requires the knowledge of a *satisfaction threshold* to represent the level of *users' satisfaction* with the proposed schedule. The common limitation of all these approaches is their assumption that the users' preferences are known (be it exactly or through the moments of their underlying distributions), which is unrealistic as these preferences are not only commonly unknown but must be elicited, which is considered to be an expensive process.

To contrast these limitations, this paper develops a scheduling system that minimizes energy consumption while modeling users' satisfaction with schedules as *expensive to evaluate* functions that can be approximately learned [12,37,36,40,20]. These functions are expensive to evaluate in the sense that they require user interactions and feedback as part of the evaluations and users may not tolerate a large number of these interactions [9]. The paper adopts Bayesian optimization [18,15] to learn those users' satisfaction that are salient to optimization of the appliances' schedules. Bayesian optimization is an iterative process suitable for optimizing expensive to evaluate functions. It consists of two main components, a Bayesian statistical method to model the objective function and an acquisition function to determine the next location to sample.

The contributions of this paper are the following: (1) It develops a constraint-based model for scheduling appliances with an expensive to evaluate user's satisfaction; (2) It introduces a Bayesian optimization-based algorithm to find schedules that maximize the user's satisfaction while limiting the interactions between the system and the user; (3) It proposes a set of customized acquisition functions for the Bayesian optimization that are tailored to our formulation; and (4) It evaluates the effectiveness of the proposed Bayesian optimization algorithm over a variety of appliances scheduling problems. The proposed customized acquisition functions improve the algorithm's performance up to 26% compared to their baseline counterparts.

In Section 2, we discuss prior work that is related to the work presented in this paper. In Section 3, we provide a brief background before we explain our problem definition and solution approaches in Sections 4 and 5, respectively. Finally, we experimentally evaluate our solution approaches in Section 6 and conclude in Section 7.

2 Related Work

Within the past decade, a wide variety of optimization approaches have emerged to model and solve the *smart appliance scheduling* (SAS) problems in residential buildings. For instance, integer linear programming [4,42], mixed integer programming [3,1], and convex optimization [31,26] techniques have been proposed to shift energy demand over a daily forecast price cycle in which the cost or energy consumption is minimized. Unlike most of these approaches that have solely focused on minimizing energy consumption or cost, our approach also considers users' comfort and preferences and proposes schedules that are more aligned with users' preferences while reducing energy cost.

Aside from our work, there also exist others who jointly optimized energy consumption and user preferences. For example, Fioretto *et al.* formulated the SAS problem as a distributed constraint optimization problem to minimize energy cost and peak load across a neighborhood of multiple smart homes, while allowing pre-defined users' constraints [8]. Zhu *et al.* used a variant of particle swarm optimization for the SAS problem to minimize energy cost and users' discomfort (opposite of users' preference/comfort) across a neighborhood of multiple smart homes, assuming a pre-defined users' comfort parameter [41]. Nguyen *et al.* formulated the SAS problem with probabilistic users' preferences–preferences for using appliances are uncertain, represented as Normal distributions. They proposed sample average approximation and depth-first search techniques to solve the SAS problem that aim at minimizing energy cost while ensuring that the user preference is within some acceptable threshold [20]. Finally, Tabakhi *et al.* used a distributed constraint optimization approach for the SAS problem to minimize energy cost and users' discomfort where constraints encode users' preferences that are represented as Normal distributions [35,34]. Different from these studies, we assume that the users' satisfaction with a schedule is unknown or expensive to evaluate, but could be approximately learned. Our proposed approach aims to find the schedule that maximizes user's satisfaction while reducing energy cost.

3 Background

This section reviews some basic notions about constraint satisfaction problems and Bayesian optimization.

3.1 Constraint Satisfaction Problems (CSPs)

The *Constraint Satisfaction Problem* (CSP) [24,23] is a powerful framework for formulating and solving combinatorial problems such as scheduling [30]. It is defined as a tuple $\mathcal{P} = \langle \mathcal{V}, \mathcal{T}, \mathcal{F} \rangle$, where $\mathcal{V} = \{v_1, \cdots, v_n\}$ is a set of n variables, $\mathcal{T} = \{T_1, \cdots, T_n\}$ is a set of domains, where each T_i is associated with variable V_i and expresses the possible values V_i can take, and $\mathcal{F} = \{f_1, \cdots, f_m\}$ is a set of m constraints, restricting the values that the variables can take simultaneously.

A constraint $f_j = \langle C(f_j), R(f_j) \rangle$ is a pair, where $C(f_j) \subseteq \mathcal{V}$ is the scope of f_j (i.e., a set of variables relevant to f_j) and $R(f_j)$ is a relation that consists of all the value combinations that the variables in $C(f_j)$ can take. The constraint f_j is *satisfied* if the variables take on one of the value combinations in $R(f_j)$ and is *unsatisfied* otherwise. The cardinality of $C(f_j)$ (i.e., the number of variables involved in f_j) is also called the constraint's arity.

A solution to a CSP is a value assignment to all variables such that all the constraints are satisfied. Solving a CSP is to find a solution of the CSP or to prove that no solution exists (i.e., no combination of value assignments satisfy all constraints).

3.2 Bayesian Optimization

Bayesian optimization [18,15] is a powerful method used to find a global maxima (or minima) of a function $\mathtt{u} : \mathtt{X} \rightarrow \mathbb{R}$

$$\max_{\mathtt{x}} \mathtt{u}(\mathtt{x}), \tag{1}$$

where $\mathtt{x} \in \mathtt{X} \subseteq \mathbb{R}^d$ [9]. The function $\mathtt{u}(.)$ typically has the following properties:

- It is an "expensive to evaluate" function: In the context of this paper, evaluating $\mathtt{u}$ at a point $\mathtt{x}$ requires interacting with the user, who will not tolerate a large number of interactions.
- It is a "black-box" function: It is a continuous function but lacks known structures, like linearity or concavity or it is a "derivative-free" function when only $\mathtt{u}(\mathtt{x})$ is observed and no first- or second-order derivatives exist [9].

Bayesian optimization is an iterative process consists of two main ingredients: A Bayesian statistical method for modeling the objective function, typically a *Gaussian Process* (GP) regression, and an *acquisition function* for deciding where to evaluate the objective function. A GP provides a posterior probability distribution of the function $\mathtt{u}(\mathtt{x})$ at a candidate point $\mathtt{x}$. Then, the acquisition function measures the value that would be generated by evaluating $\mathtt{u}$ at a new point $\mathtt{x}'$, based on the current posterior distribution over $\mathtt{u}$. Every time that the function is evaluated at a new point, the posterior distribution is updated [9]. We briefly discuss GPs regression and acquisition functions in the following.

Gaussian Process (GP) Regression [43,17] is a Bayesian statistical approach for modeling functions [9]. A GP provides a distribution over functions $\mathtt{u}(\mathtt{x}) \sim \mathcal{GP}(\mu(\mathtt{x}), \mathtt{k}(\mathtt{x}, \mathtt{x}'))$, specified by its mean function, μ and covariance (kernel) function $\mathtt{k}$, for any pairs of input points $\mathtt{x}$, and $\mathtt{x}'$ (where $\mathtt{x}, \mathtt{x}' \in \mathtt{X}$). Intuitively, a GP (i.e., $\mathcal{GP}(.,.)$) is similar to a function, except that instead of returning a scalar $\mathtt{u}(\mathtt{x})$, it returns the mean and standard deviation of a normal distribution over the possible values of $\mathtt{u}$, at any arbitrary input point $\mathtt{x}$. For convenience, we assume that the prior mean function is the zero mean function $\mu(\mathtt{x}) = 0$. The kernel function $\mathtt{k}(\mathtt{x}, \mathtt{x}')$ describes the covariance of a GP random variables. The mean function and the kernel function together define a GP. Some of the commonly used kernel functions are the radial basis function (RBF) [38], Matérn [38], and periodic [5] kernels. In the context of this paper, we choose the periodic kernel function as the default kernel function and show the behavior of the algorithm using Matérn and RBF kernels in Section 6.3.

Acquisition Functions use a GP model to guide the selection of a global maxima of the objective function $\mathtt{u}$, by proposing sampling points in the search space. The acquisition function provides a trade-off between exploration and exploitation. Exploration means sampling (i.e., evaluating the objective function) at locations where the GP model has higher uncertainty and exploitation means that sampling at locations where the GP model has higher estimated values. Both exploration and exploitation correspond to high acquisition values. Determining where to evaluate next is achieved by maximizing the acquisition function.

More formally, the function $\mathtt{u}$ will be evaluated at $\mathtt{x}^+ = \text{argmax}_{\tilde{\mathtt{x}}}\, \alpha(\tilde{\mathtt{x}})$ given a set of sampling points drawn from $\mathtt{u}$, where α is the acquisition function. We will discuss standard acquisition functions in more detail in Section 5.

4 The Smart Appliance Scheduling (SAS) Problem with Users' Satisfaction

We describe the *Smart Appliances Scheduling* (SAS) problem and model it as a CSP, given the constraints imposed by users on the appliances. Our model is similar to the one defined in [20], except that we do not represent the users' preference as Normal distributions when we model the SAS problem. We first find all the SAS problem's feasible schedules using a CSP solver. Then, we assign a user's satisfaction to each schedule. Among all feasible schedules, our algorithm aims at finding the schedule that maximizes user's satisfaction.

Definition 1. *The* Smart Appliances Scheduling *problem P is a tuple $\langle \mathcal{V}, \mathcal{T}, \mathcal{F} \rangle$, where*

- $\mathcal{V} = \{v_1, \cdots, v_n\}$ *is a finite set of smart appliances.*
- $\mathcal{T} = \{T_1, \cdots, T_n\}$ *is a set of domains of smart appliances, where* $T_j = \{1, \cdots, \mathcal{H}\}$ *is a set of time slots for appliance* $v_j \in \mathcal{V}$*, and* $\mathcal{H}$ *is a finite time horizon. Each appliance* v_j *must choose a* start time $t_k \in T_j$ *to turn on.*
- $\mathcal{F} = \mathcal{F}_1 \cup \mathcal{F}_2$ *is a set of constraints, where*

$$\mathcal{F}_1 = \{\langle v_j, (\{Before, After, At\}, t_k)\rangle\} \tag{2}$$

is a set of unary constraints indicating that appliance v_j *must be turned on either before, after, or at time* $t_k \in T$*; and*

$$\mathcal{F}_2 = \{\langle (v_i, v_j), \{Before, After, Parallel\}\rangle\} \tag{3}$$

for some pairs of variables $v_i, v_j \in \mathcal{V}$ (with $v_i \neq v_j$), is a set of binary constraints indicating that appliance v_i must be turned on either before, after, or at the same time that appliance v_j is turned on. We refer to these binary constraints as dependencies between appliances.

A candidate solution for P is a schedule $\mathtt{x} = \{(v_1, t_1), \cdots, (v_n, t_n)\}$ that consists of the set of all appliances v_j and their corresponding start times t_k. A solution for P is a schedule that satisfies all the unary and binary constraints. We also use the term *feasible schedules* to refer to solutions. The set of all feasible schedules is denoted by $\mathbf{X}$.

Given the set of feasible schedules $\mathbf{X}$, we are only interested in the schedules that maximize a user's satisfaction as the user can prefer one schedule over the other. The user's satisfaction with the schedule depends on various components. For example, temperature, the price of energy (electricity), the time of day, and personal concerns might affect the user's satisfaction [21,29,41].

4.1 The User's Satisfaction Function

We define a user's satisfaction function that is composed of two components similar to [29,41] as follows:

$$\mathtt{u}_{v_j}^{t_k} = \left| \left(\frac{\alpha_c}{1 + e^{-c(\zeta, (v_j, t_k))}} - \frac{\alpha_e}{1 + e^{-\varepsilon(p, (v_j, t_k))}} \right) \right| \tag{4}$$

where:

- α_e and α_c are non-negative weights values such that $\alpha_e + \alpha_c = 1$.
- $c : \mathbb{R} \times \mathcal{V} \times \mathcal{T} \rightarrow \mathbb{R}$ is a *comfort function* (defined below) with $c(\zeta, (v_j, t_k))$ quantifying the comfort level of the users for appliance v_j to turn on at time t_k, given the user's comfort ζ.
- $\varepsilon : \mathbb{R} \times \mathcal{V} \times \mathcal{T} \rightarrow \mathbb{R}$ is an *energy cost function* (defined below) with $\varepsilon(p, (v_j, t_k))$ quantifying the cost of energy for appliance v_j to turn on at time t_k, given a real-time pricing schema p.

The comfort c is defined as:

$$c(\zeta, (v_j, t_k)) = \sum_{i=t_k}^{t_k + \delta_{v_j}} \varsigma_i \cdot \sigma \tag{5}$$

where $\varsigma_i \in \zeta$ is the user's comfort at time i, $\sigma \in [0, 1]$ is a parameter that determines user's sensitivity to any change in time, and δ_{v_j} is the duration of v_j (i.e., the period of time that v_j must run until it finishes its task without any interruption).

The cost of energy assumes that the energy provider of each smart home sets energy prices according to a real-time pricing schema p specified at regular intervals $t_k \in T$. The energy cost function is defined as:

$$\varepsilon(p, (v_j, t_k)) = \sum_{i=t_k}^{t_k + \delta_{v_j}} p_i \cdot \rho_{v_j} \tag{6}$$

where $p_i \in p$ is the price of energy at time i, ρ_{v_j} is the power consumption of appliance v_j at each time step, and δ_{v_j} is the duration of v_j.

Definition 2. *The user satisfaction on the dependency of variables v_i and v_j appearing in $\mathcal{F}_2$ is defined as*

$$\mathtt{u}_{(v_i,v_j)}^{(t_l,t_k)} = \mathtt{u}_{v_i}^{t_l} + \mathtt{u}_{v_j}^{t_k}$$

Definition 3. *The user's satisfaction of the schedule* $\mathtt{x}$ *is:*

$$\mathtt{u}(\mathtt{x}) = \frac{1}{|\mathcal{V}|}\left(\sum_{v_m \in \mathcal{F}_1} \mathtt{u}_{v_m}^{t_o} + 2 \sum_{(v_i,v_j) \in \mathcal{F}_2} \mathtt{u}_{(v_i,v_j)}^{(t_l,,t_k)}\right) \tag{7}$$

The user's satisfaction with the suggested schedule is inherently subjective and varies from one individual to another. Due to a large number of feasible schedules for smart home appliances within a day, eliciting user's satisfaction for every schedule is cumbersome as users might not tolerate an unlimited number of queries. Thus, we presume that the user's satisfaction is an expensive to evaluate function and employ the Bayesian optimization strategy to optimize the function and find the best possible schedule $\mathtt{x}^*$:

$$\mathtt{x}^* = \underset{\mathtt{x}}{\operatorname{argmax}}\, \mathtt{u}(\mathtt{x}). \tag{8}$$

with only a limited number of queries to ask the user their satisfaction.

5 Bayesian Optimization for the SAS Problem

Recall that Bayesian optimization uses a Gaussian process to approximate the user's satisfaction function (defined in Definition 3) and an acquisition function to iteratively identify the next candidate schedule to evaluate given the observations (i.e., a set of candidate schedules and their corresponding users' satisfaction) thus far. Typically, an initial observation of at least two candidate schedules and their corresponding users' satisfaction is assumed to be available, denoted by $\mathcal{D}_0$ in Algorithm 1. Given the initial observation, the acquisition function is used to identify the next and subsequent candidate schedules to evaluate.

In this paper, we use an off-the-shelf CSP solver to find the set of all feasible schedules $\mathbf{X}$. Then, to generate the initial k observations, we randomly choose a set of k schedules from $\mathbf{X}$ and evaluate them using Eq. 7. After using the acquisition function to identify the next candidate schedule and evaluating it (i.e., eliciting the user's satisfaction with the candidate schedule), the prior is updated to produce a more informed posterior distribution over the user's satisfaction function. Algorithm 1 illustrates the above process.

Algorithm 1 Bayesian Optimization for the SAS problem

Input: The set of all feasible schedules $\mathbf{X}$, the number of initial observation k, an acquisition function α, and the number of queries Q
Output: The schedule with the highest user's satisfaction after Q queries
  Select k schedules from $\mathbf{X}$ randomly as the initial observation set $\mathcal{D}_0$
  Construct the GP model on $\mathcal{D}_0$
  for $n = 0, 1, 2, \cdots, Q$ **do**
    Find $\mathtt{x}_{n+1}$ by maximizing the acquisition function α: $\mathtt{x}_{n+1} = \text{argmax}_{\tilde{\mathtt{x}}}\, \alpha(\tilde{\mathtt{x}})$
    Elicit the user's satisfaction for schedule $\mathtt{x}_{n+1} : y_{n+1} \leftarrow \mathtt{u}(\mathtt{x}_{n+1})$
    Augment the observation set $\mathcal{D}_{n+1} = \mathcal{D}_n \cup \{(x_{n+1}, y_{n+1})\}$
    Re-construct the GP model of $\mathtt{u}$ on the augmented set $\mathcal{D}_{n+1}$
  end for

5.1 Existing Acquisition Functions

We briefly review some popular acquisition functions and adapt them to the SAS problem.

Expected Improvement (EI): Originally proposed by Mockus [16], the function tries to balance between exploration (evaluating at points with higher uncertainty) versus exploitation (evaluating at points with higher estimated values) and is defined as:

$$EI(\mathtt{x}) = \Delta(\mathtt{x}, \mathcal{F}_1, \mathcal{F}_2) \cdot \Big(\big(\mu(\mathtt{x}) - \mathtt{u}(\mathtt{x}^+) - \xi\big)\, \Phi(Z) + \sigma(\mathtt{x})\phi(Z) \Big), \tag{9}$$

where $Z = \frac{\mu(\mathtt{x}) - \mathtt{u}(\mathtt{x}^+) - \xi}{\sigma(\mathtt{x})}$, $\mathtt{u}(\mathtt{x}^+)$ is the value of the best observation so far, $\mathtt{x}^+$ is the schedule with that observation, and finally:

$$\Delta(\mathtt{x}, \mathcal{F}_1, \mathcal{F}_2) = \begin{cases} 1 & \text{if } \mathbf{x} \text{ satisfies all constraints in } \mathcal{F}_1 \text{ and } \mathcal{F}_2 \\ 0 & \text{otherwise,} \end{cases} \tag{10}$$

is the schedule satisfiability indicator function with $\mathcal{F}_1$ and $\mathcal{F}_2$ denoting the sets of constraints over the appliances of an individual' smart home. The above assigns zero improvement to all unsatisfiable schedules. The mean and standard deviation of the GP posterior, predicted at $\mathtt{x}$ are denoted as $\mu(\mathtt{x})$ and $\sigma(\mathtt{x})$, respectively. Additionally, Φ and ϕ denote the standard cumulative density function and probability density function, respectively. Finally, ξ is a parameter that trades exploration for exploitation, where higher ξ leads to more exploration.

Probability of Improvement (PI): An alternative function to EI [44,11], it is extended to be a constrained improvement function as follows:

$$PI(\mathtt{x}) = \Delta(\mathtt{x}, \mathcal{F}_1, \mathcal{F}_2) \cdot \Phi(Z) \tag{11}$$

Upper Confidence Bound (UCB): Adopted for Bayesian optimization [32], it is extended to be a constrained acquisition function as follows:

$$UCB(\mathtt{x}) = \Delta(\mathtt{x}, \mathcal{F}_1, \mathcal{F}_2) \cdot (\mu(\mathtt{x}) + \kappa\sigma(\mathtt{x})) \tag{12}$$

where κ is the parameter that trades off exploration and exploitation.

5.2 Acquisition Functions for the SAS Problem

We propose a set of acquisition functions that are extended from the existing functions above and are customized for our SAS problem. Specifically, These customized acquisition functions employ the energy cost function of the candidate schedule, defined below, as a contributing factor.

Definition 4. *The energy cost function of the candidate schedule* $\mathtt{x}$*, is the total energy cost of scheduling all the appliances involved in* $\mathtt{x}$*:*

$$\varepsilon(p, \mathtt{x}) = \sum_{(v_j, t_k) \in \mathtt{x}} \varepsilon(p, (v_j, t_k)) \tag{13}$$

and $\varepsilon(p, (v_j, t_k))$ *is the energy cost function defined in Eq. 6.*

In the following, we introduce the acquisition functions, where the energy cost function of the candidate schedule is used as a *trade-off parameter* to provide a better balance between exploration and exploitation during the optimization process.

EI with Energy Cost as a Trade-off Parameter (EI-ECT): The proposed acquisition function exploits the energy cost function $\varepsilon(p, \mathtt{x})$ of the candidate schedule $\mathtt{x}$ as the trade-off parameter in EI:

$$\text{EI-ECT}(\mathtt{x}) = \Delta(\mathtt{x}, \mathcal{F}_1, \mathcal{F}_2) \cdot \Big(\big(\mu(\mathtt{x}) - \mathtt{u}(\mathtt{x}^+) - \varepsilon(p, \mathtt{x})\big) \Phi(Z_{ECT}) + \sigma(\mathtt{x}) \phi(Z_{ECT}) \Big) \tag{14}$$

where $Z_{ECT} = \frac{\mu(\mathtt{x}) - \mathtt{u}(\mathtt{x}^+) - \varepsilon(p, \mathtt{x})}{\sigma(\mathtt{x})}$, and $\mathtt{u}(\mathtt{x}^+)$ is the value of the best observation so far.

PI with Energy Cost as a Trade-off Parameter (PI-ECT): The proposed acquisition function exploits the energy cost function $\varepsilon(p, \mathtt{x})$ as a trade-off parameter in PI:

$$\text{PI-ECT}(\mathtt{x}) = \Delta(\mathtt{x}, \mathcal{F}_1, \mathcal{F}_2) \cdot \Phi(Z_{ECT}) \tag{15}$$

UCB with Energy Cost as a Trade-off Parameter (UCB-ECT): The proposed acquisition function exploits the energy cost function $\varepsilon(p, \mathtt{x})$ as a trade-off parameter in UCB:

$$\text{UCB-ECT}(\mathtt{x}) = \Delta(\mathtt{x}, \mathcal{F}_1, \mathcal{F}_2) \cdot \Big(\mu(\mathtt{x}) + \varepsilon(p, \mathtt{x}) \sigma(\mathtt{x}) \Big) \tag{16}$$

In the following, we introduce the acquisition functions, where the energy cost function of the candidate schedule is used as a *coefficient parameter* to improve the correlation between the acquisition function and the objective function $\mathtt{u}(\cdot)$ during the optimization process.

EI with Energy Cost as a Coefficient (EI-ECC): The proposed acquisition function exploits the energy cost function $\varepsilon(p, \mathtt{x})$ of the candidate schedule $\mathtt{x}$ as a coefficient in EI:

$$\text{EI-ECC}(\mathtt{x}) = \Delta(\mathtt{x}, \mathcal{F}_1, \mathcal{F}_2) \cdot \Big(\big(\mu(\mathtt{x}) - \mathtt{u}(\mathtt{x}^+) - \xi\big) \Phi(Z) + \sigma(\mathtt{x})\phi(Z)\Big) \cdot \varepsilon(p, \mathtt{x}) \tag{17}$$

PI with Energy Cost as a Coefficient (PI-ECC): The proposed acquisition function exploits the energy cost function $\varepsilon(p, \mathtt{x})$ as a coefficient in PI:

$$\text{PI-ECT}(\mathtt{x}) = \Delta(\mathtt{x}, \mathcal{F}_1, \mathcal{F}_2) \cdot \Phi(Z) \cdot \varepsilon(p, \mathtt{x}) \tag{18}$$

UCB with Energy Cost as a Coefficient (UCB-ECC): The proposed acquisition function exploits the energy cost function $\varepsilon(p, \mathtt{x})$ as a coefficient in UCB:

$$\text{UCB-ECC}(\mathtt{x}) = \Delta(\mathtt{x}, \mathcal{F}_1, \mathcal{F}_2) \cdot \big(\mu(\mathtt{x}) + \kappa\sigma(\mathtt{x})\big) \cdot \varepsilon(p, \mathtt{x}) \tag{19}$$

6 Empirical Evaluations

We empirically evaluate our new acquisition functions (i.e., EI-ECT, EI-ECC, PI-ECT, PI-ECC, UCB-ECT, and UCB-ECC) by comparing them against three popular acquisition functions (i.e., EI, PI, and UCB) that serve as baseline acquisition functions. Specifically, we compare their *loss*, defined as $\mathtt{u}_{max} - \mathtt{u}(\mathtt{x}^*)$, where $\mathtt{u}_{max}$ is the maximal value of the user's satisfaction, $\mathtt{x}^*$ is the best schedule elicited after a certain number of queries, and $\mathtt{u}(\mathtt{x}^*)$ is the user's satisfaction with that schedule [39]. To evaluate the performance of the Bayesian optimization algorithm with different acquisition functions, we report the loss averaged over 20 SAS problem instances with 10 runs per instance.

6.1 Experimental Setup

In all experiments, we set the time horizon $\mathcal{H} = 24$ with $|T| = 48$ time steps and vary the number of smart appliances $|\mathcal{V}| = [2, 5]$. Our focus is on scheduling the smart appliances that accept flexible schedules and are more likely to be available in typical homes. For instance, Tesla electric vehicle, Kenmore oven and dishwasher, GE clothes washer and dryer, iRobot vacuum cleaner, and LG air conditioner [8] can operate at any time step to finish their tasks. We randomly generated a set of unary constraints $\mathcal{F}_1$ enforced on appliances, a set of binary constraints $\mathcal{F}_2$ between appliances in each home and set the number of binary constraints $|\mathcal{F}_2| = p_1 \cdot \frac{(|\mathcal{V}|)(|\mathcal{V}|-1)}{2}$, where $p_1 = 0.4$.

We used MiniZinc [33], an off-the-shelf CSP solver, to solve the SAS problem instances and find the set of all feasible solutions $\mathbf{X}$. As defined earlier, the user's satisfaction is a function of energy cost and user's comfort given a schedule, which we describe below.

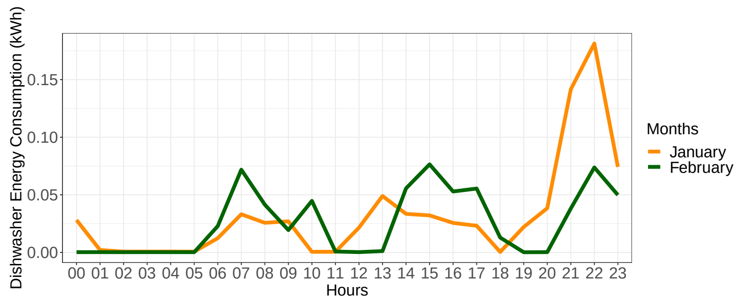


Fig. 1. Averaged hourly energy consumption in kWh for the dishwasher in January and February of 2018 for a single household in Texas. The data is collected by Pecan Street Inc.

Real-time Pricing Schema: We adapted the pricing schema defined by California ISO (CAISO),[3] which includes the energy price (cents/kWh) at a granularity of 5-minute intervals. To create the hourly real-time pricing schema p, we calculated the price by averaging over twelve data points in each hour and converting the price from cents to dollars. The duration of the task δ_{v_j} of each appliance v_j is set to 2 hours.

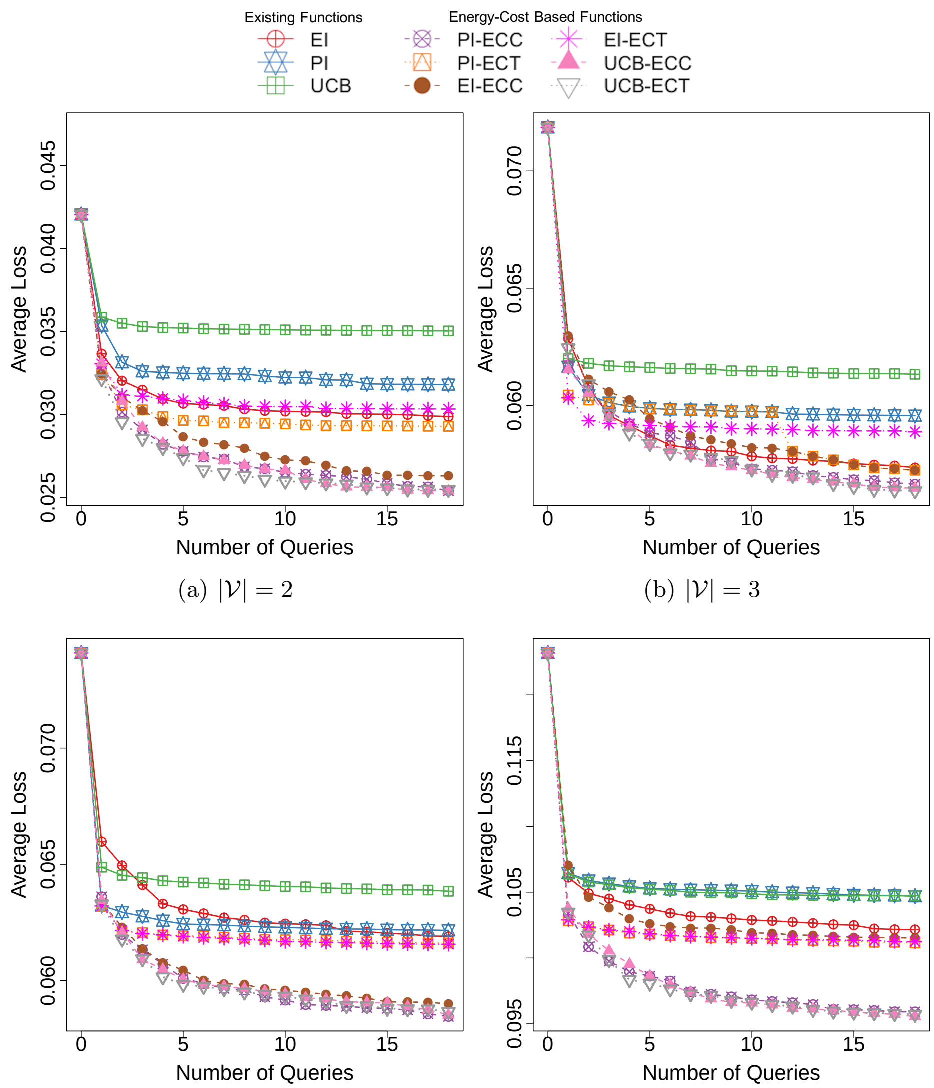
User's Comfort: To create the user's comfort level ζ, we use the users load profile data collected by Pecan Street Inc [22].[4] The time-series energy datasets consist of energy consumption of multiple appliances in kWh that have been collected every 15 minutes for 25 homes in a region in Texas and 25 homes in California in 2018. For each of these 50 homes, we calculated each appliance's hourly energy consumption by summing up the four data points in each hour. For each of these appliances and hour of a day, we averaged the calculated hourly energy consumption over a one-month period, resulting in a total of 24 data points (per month per appliance). We assume that the user's comfort level corresponds to their daily load profile for each appliance.

Fig. 1 illustrates the averaged energy consumption of the dishwasher at every hour of January and February in 2018 for one of the homes in Texas. From the figure, we can perceive, with a high probability, that the user preferred to turn on the dishwasher at night—the most preferred time was 10:00 p.m. in January 2018 as shown with the solid orange line in the figure. Then, we normalized these hourly values for each appliance and used them as the user's comfort ζ for that appliance. In all experiments, we set the number of schedules $k = 2$ in our initial observation and we vary the number of queries Q from 1 to 18.

6.2 Impact of Energy-Cost Based Acquisition Functions

Fig. 2 shows the performance of different acquisition functions, where we report the average loss for each acquisition function. All acquisition functions were run

[3] California ISO (CAISO). https://tinyurl.com/y8t4xa2r
[4] Pecan Street Inc. https://www.pecanstreet.org/dataport/



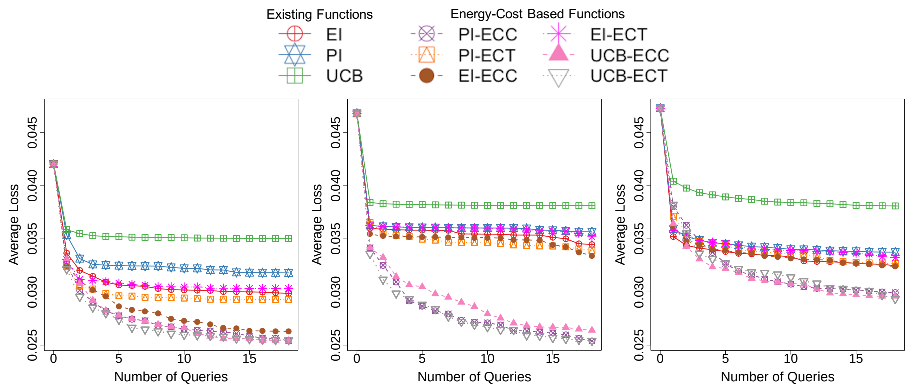
(a) $|\mathcal{V}| = 2$ (b) $|\mathcal{V}| = 3$

(c) $|\mathcal{V}| = 4$ (d) $|\mathcal{V}| = 5$

Fig. 2. The Average Loss of Energy-Cost Based Acquisition Functions Against the Existing Acquisition Functions

with identical initial observations. As expected, for all acquisition functions, the losses decreased with increasing number of queries Q. In general, we observe that our energy-cost based acquisition functions outperform their traditional baseline counterparts.

The reason EI-ECT, PI-ECT, and UCB-ECT (i.e., the energy-cost based acquisition functions that use the energy cost function as a trade-off parameter)



(a) Periodical Kernel (b) Matérn Kernel (c) RBF Kernel

Fig. 3. Effect of Kernel Functions on the Performance of the Bayesian Optimization

outperform their baseline counterparts is the following: While EI, PI, and UCB acquisition functions give rise to distinct sampling behavior over time, the exact choice of the trade-off parameters ξ and κ is left to the users. As suggested by Lizotte [13], we set $\xi = 0.01$ and $\kappa = 0.01$. Due to the high sensitivity of PI and UCB to the choice of the trade-off parameter, they performed worse than EI and other energy-cost based acquisition functions. For the acquisition functions that use energy cost as the trade-off parameter, the value of the trade-off parameter changes according to the energy cost of the selected schedule ($\varepsilon(p, \mathtt{x}) > 0.01$), allowing it to explore for more schedules with low energy costs and exploit fewer schedules with high energy costs.

The reason EI-ECC, PI-ECC, and UCB-ECC (i.e., the energy-cost based acquisition functions that use the energy cost function as a coefficient) outperform their baseline counterparts is the following: While the trade-off parameters ξ and κ are set to their default values of 0.01, the value of the coefficient parameter changes according to the energy cost of the selected schedule, affecting the GP posterior's mean $\mu(\mathtt{x})$ and standard deviation $\sigma(\mathtt{x})$. It results in evaluations of the schedules in regions with: (i) Higher uncertainty when the standard deviation is larger than the mean, and (ii) Higher estimated values when the mean is larger than the standard deviation. The former explores schedules with lower energy costs more than those with higher energy costs, and the latter exploits schedules with lower energy costs more than those with higher energy costs.

These results demonstrate the energy-cost based functions that outperformed their baseline counterparts improve the Bayesian optimization algorithm's performance up to 26% by reducing the average loss.

6.3 Impact of Different Kernel Functions

In our next experiment, to evaluate the choice of kernel function, we employed three different commonly-used kernels (periodic [5], Matérn [38], and RBF [28,38]), where we set the number of appliances $|\mathcal{V}| = 2$. Figure 3 shows the performance of the acquisition functions using different kernels, and the same trends from earlier experiments apply here as well.

7 Conclusions

The autonomous and coordinated control of smart appliances in residential buildings represents an effective solution to reduce peak demands. This paper, studied the associated *smart appliance scheduling* (SAS) problem that minimizes energy peaks while also complying with users' satisfaction about schedules. It focused on the realistic case when these users' satisfaction are unknown and, therefore, must be learned through an expensive elicitation process. To minimize the amount of elicitation requests, the proposed approach employs a *Bayesian optimization* (BO) framework, that is used to approximate a function of the user comfort and energy cost of the given schedule. The paper proposed a number of acquisition functions, which are a central component of BO as they drive the selection of desirable schedules to be evaluated (a.k.a. elicited). The proposed acquisition functions are SAS-specific and exploit the concept of *energy cost*, that encodes information about schedules cost. The experimental evaluations demonstrated the effectiveness of the proposed energy-cost based acquisition functions. Our model and solution approach thus extend the state of the art in Bayesian optimization to better formulate and solve agent-based applications with user preferences.

Acknowledgments

Tabakhi and Yeoh are partially supported by NSF grants 1550662 and 1812619, and Fioretto is partially supported by NSF grant 2007164. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the sponsoring organizations, agencies, or the U.S. government.

References

1. Amini, M.H., Frye, J., Ili, M.D., Karabasoglu, O.: Smart residential energy scheduling utilizing two stage mixed integer linear programming. In: North American Power Symposium (2015)
2. Chen, L., Pu, P.: Survey of preference elicitation methods. Tech. rep., Swiss Federal Institute of Technology in Lausanne (EPFL) (2004)
3. Chen, Z., Wu, L., Fu, Y.: Real-time price-based demand response management for residential appliances via stochastic optimization and robust optimization. IEEE Transactions on Smart Grid **3**(4), 1822–1831 (2012)
4. Du, P., Lu, N.: Appliance commitment for household load scheduling. IEEE transactions on Smart Grid **2**(2), 411–419 (2011)
5. Duvenaud, D.: The kernel cookbook. Advice on Covariance functions pp. 12–39 (2015)

6. Fang, X., Misra, S., Xue, G., Yang, D.: Smart gridthe new and improved power grid: A survey. IEEE communications surveys & tutorials **14**(4), 944–980 (2011)
7. Farhangi, H.: The path of the smart grid. IEEE power and energy magazine **8**(1), 18–28 (2009)
8. Fioretto, F., Yeoh, W., Pontelli, E.: A multiagent system approach to scheduling devices in smart homes. In: Proceedings of AAMAS. pp. 981–989 (2017)
9. Frazier, P.I.: A tutorial on bayesian optimization. CoRR **abs/1807.02811** (2018)
10. Gelazanskas, L., Gamage, K.A.: Demand side management in smart grid: A review and proposals for future direction. Sustainable Cities and Society **11**, 22–30 (2014)
11. Jones, D.R.: A taxonomy of global optimization methods based on response surfaces. Journal of global optimization **21**(4), 345–383 (2001)
12. Le, T., Tabakhi, A.M., Tran-Thanh, L., Yeoh, W., Son, T.C.: Preference elicitation with interdependency and user bother cost. In: Proceedings of AAMAS. pp. 1459–1467 (2018)
13. Lizotte, D.J.: Practical bayesian optimization. University of Alberta (2008)
14. Logenthiran, T., Srinivasan, D., Shun, T.Z.: Demand side management in smart grid using heuristic optimization. IEEE transactions on smart grid **3**(3), 1244–1252 (2012)
15. Mockus, J., Mockus, L.: Bayesian approach to global optimization and application to multiobjective and constrained problems. Journal of Optimization Theory and Applications **70**(1), 157–172 (1991)
16. Mockus, J.: On bayesian methods for seeking the extremum and their application. In: Proceedings of Information Processing. pp. 195–200 (1977)
17. Mockus, J.: Application of bayesian approach to numerical methods of global and stochastic optimization. Journal of Global Optimization **4**(4), 347–365 (1994)
18. Mockus, J., Tiesis, V., Zilinskas, A.: The application of bayesian methods for seeking the extremum. Towards global optimization **2**(117-129), 2 (1978)
19. Mohsenian-Rad, A.H., Wong, V.W., Jatskevich, J., Schober, R., Leon-Garcia, A.: Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid. IEEE transactions on Smart Grid **1**(3), 320–331 (2010)
20. Nguyen, V., Yeoh, W., Son, T.C., Kreinovich, V., Le, T.: A scheduler for smart homes with probabilistic user preferences. In: Proceedings of PRIMA. pp. 138–152 (2019)
21. Peeters, L., De Dear, R., Hensen, J., Dhaeseleer, W.: Thermal comfort in residential buildings: Comfort values and scales for building energy simulation. Applied energy **86**(5), 772–780 (2009)
22. Rhodes, J.D., Upshaw, C.R., Harris, C.B., Meehan, C.M., Walling, D.A., Navrátil, P.A., Beck, A.L., Nagasawa, K., Fares, R.L., Cole, W.J., et al.: Experimental and data collection methods for a large-scale smart grid deployment: Methods and first results. Energy **65**, 462–471 (2014)
23. Rossi, F., van Beek, P., Walsh, T. (eds.): Handbook of Constraint Programming. Elsevier (2006)
24. Rossi, F., Petrie, C.J., Dhar, V.: On the equivalence of constraint satisfaction problems. In: European Conference on Artificial Intelligence. pp. 550–556 (1990)
25. Rust, P., Picard, G., Ramparany, F.: Using message-passing DCOP algorithms to solve energy-efficient smart environment configuration problems. In: Proceedings of IJCAI. pp. 468–474 (2016)
26. Samadi, P., Mohsenian-Rad, H., Schober, R., Wong, V.W.: Advanced demand side management for the future smart grid using mechanism design. IEEE Transactions on Smart Grid **3**(3), 1170–1180 (2012)

27. Scott, P., Thiébaux, S., Van Den Briel, M., Van Hentenryck, P.: Residential demand response under uncertainty. In: International Conference on Principles and Practice of Constraint Programming. pp. 645–660 (2013)
28. Sedghi, M., Atia, G., Georgiopoulos, M.: Kernel coherence pursuit: A manifold learning-based outlier detection technique. In: Proceedings of ACSSC. pp. 2017–2021 (2018)
29. Shann, M., Seuken, S.: An active learning approach to home heating in the smart grid. In: Proceedings of IJCAI. pp. 2892–2899 (2013)
30. Song, W., Kang, D., Zhang, J., Cao, Z., Xi, H.: A sampling approach for proactive project scheduling under generalized time-dependent workability uncertainty. Journal of Artificial Intelligence Research **64**, 385–427 (2019)
31. Sou, K.C., Weimer, J., Sandberg, H., Johansson, K.H.: Scheduling smart home appliances using mixed integer linear programming. In: Proceedings of CDC. pp. 5144–5149 (2011)
32. Srinivas, N., Krause, A., Kakade, S.M., Seeger, M.W.: Gaussian process optimization in the bandit setting: No regret and experimental design. In: Proceedings of ICML. pp. 1015–1022 (2010)
33. Stuckey, P.J., Becket, R., Brand, S., Brown, M., Feydy, T., Fischer, J., de la Banda, M.G., Marriott, K., Wallace, M.: The evolving world of MiniZinc. Constraint Modelling and Reformulation pp. 156–170 (2007)
34. Tabakhi, A.M.: Preference elicitation in dcops for scheduling devices in smart buildings. In: Proceedings of AAAI. pp. 4989–4990 (2017)
35. Tabakhi, A.M., Le, T., Fioretto, F., Yeoh, W.: Preference elicitation for dcops. In: Proceedings of CP. pp. 278–296 (2017)
36. Tabakhi, A.M., Yeoh, W., Yokoo, M.: Parameterized heuristics for incomplete weighted csps with elicitation costs. In: Proceedings of AAMAS. pp. 476–484 (2019)
37. Truong, N.C., Baarslag, T., Ramchurn, S.D., Tran-Thanh, L.: Interactive scheduling of appliance usage in the home. In: Proceedings of IJCAI. pp. 869–877 (2016)
38. Williams, C.K., Rasmussen, C.E.: Gaussian processes for machine learning, vol. 2. MIT press Cambridge, MA (2006)
39. Wilson, J., Hutter, F., Deisenroth, M.: Maximizing acquisition functions for bayesian optimization. In: Advances in Neural Information Processing Systems (2018)
40. Xiao, Y., Tabakhi, A.M., Yeoh, W.: Embedding preference elicitation within the search for DCOP solutions. In: Proceedings of AAMAS. pp. 2044–2046 (2020)
41. Zhu, J., Lin, Y., Lei, W., Liu, Y., Tao, M.: Optimal household appliances scheduling of multiple smart homes using an improved cooperative algorithm. Energy **171**, 944–955 (2019)
42. Zhu, Z., Tang, J., Lambotharan, S., Chin, W.H., Fan, Z.: An integer linear programming based optimization for home demand-side management in smart grid. In: IEEE PES Innovative Smart Grid Technologies Conference. pp. 1–5 (2012)
43. Žilinskas, A.: On the use of statistical models of multimodal functions for the construction of the optimization algorithms. In: Optimization techniques, pp. 138–147. Springer (1980)
44. Žilinskas, A.: A review of statistical models for global optimization. Journal of Global Optimization **2**(2), 145–153 (1992)