2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

Integer-free Optimal Scheduling of Smart Appliances

Felix Petzke and Stefan Streif

***Abstract*— With the growing number of smart appliances in the electrical grid, flexible optimization techniques capable of scheduling large scale systems of such devices are needed in order to control them optimally. However, scheduling approaches in the literature predominantly use mixed-integer formulations with fixed time slots. This usually yields a poor scalability due to the computational complexity of the resulting mixed-integer optimization problems. Furthermore this approach can be rather inflexible since it restricts the task executions to specific time intervals. This may become problematic if the (actual) optimal task execution would only span a fraction of one time slot, as it might happen for applications with very short individual tasks.**

This paper presents an optimization based scheduling approach that creates a *continuous* schedule, i.e. the task execution is not bound to specific time slots. Furthermore, no integer variables are used in the process. This makes it not only applicable to processes with arbitrarily short execution time lengths, but also has the potential of a reduced computational complexity compared to mixed-integer based approaches. Preliminary case studies demonstrate the advantages of the proposed approach.

***Index Terms*— optimization, scheduling, wrap around, smart appliances**

I. INTRODUCTION

With the rising popularity of smart grid technologies like smart appliances and smart meters in residential environments, the problem of optimal scheduling gains importance in the high performance control of such systems [1], [14].

This paper is concerned with the cost optimal scheduling of smart appliances and storage devices that form a nanogrid, following the definition in [13]. The considered appliances are assumed to have arbitrarily short task execution times and discrete power levels (cf. Section II). The optimization is performed with regard to the external power price profile, which is assumed to be known.

The question of when to perform a task (e.g. charging a battery) has been raised in the fault tolerant [15] and robust [14] optimal control of nanogrid systems. Many of such optimization-based control approaches for smart grids use binary variables due to their straight-forward implementation of time-based constraints like minimum or maximum execution times. This formulation results in a mixed integer program (MIP) as the optimization problem [4], [5], [10], [14], [15], [19]. While there are fast off the shelf solvers like CPLEX, GLPK, or COIN-OR for MIPs available, the problem complexity and computational demand usually still scales poorly with increasing problem dimensions. This is generally undesirable in online optimal control setups like MPC since the available time to make a decision is limited [3], [14], [15]. Therefore, many MIP approaches are performed either offline or are only applicable to small or medium sized problems. Other approaches based on real time pricing use demand response management [7], [11], [20], game theory [2], [16], or Markov chains [8] in order to derive a schedule.

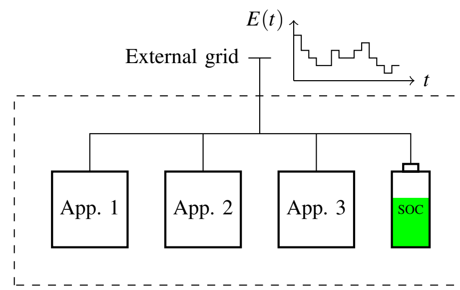

Fig. 1. Exemplary nano grid consisting of smart appliances App. 1–3 and a battery storage with an associated state of charge (SOC). The external power price profile $E(t)$ can be obtained from publicly available platforms like EPEXSPOT.

All of the mentioned approaches have in common that they use fixed time slots, i.e. the resulting schedule will only have the "resolution" of the respective time intervals chosen (usually 15 minutes [14], [15]). This can become problematic if scheduled tasks have arbitrarily short execution times. A reduction of the time window size in order to increase the resolution would again increase the number of involved decision variables and therefore the computational complexity.

The approach taken in this paper allows to schedule smart appliances (see Section II for the considered system components) *continuously* in time, i.e. the execution of a task may start at any time during the scheduling horizon and is not bound to the start of specific time slots. The same is true for a switching between operation modes. This gained flexibility has the potential to yield a lower cost solution and is also capable of including tasks that are shorter than the length of the time slots. The main idea is an integration-based reformulation of the binary optimization problem (which is presented in Section III), as shown in Section IV. The resulting problem is then split into two subproblems in order to solve it efficiently, followed by a task ordering step based on McNaughtons wrap-around algorithm [12] that produces the actual schedule. The split-up as well as the task ordering

This work was part of the research project SyNErgIt and funded by the European Union, European Social Fund ESF, Saxony.

F. Petzke and S. Streif are with the Automatic Control & System Dynamics Lab at the Technische Universität Chemnitz, Faculty for Electrical Engineering and Information Technology, 09126 Chemnitz {felix.petzke, stefan.streif}@etit.tu-chemnitz.de.

are presented and discussed in Section V.

The resulting optimization problems use no integer variables anymore. This point is particularly important in the scheduling of large- and very-large-scale systems of appliances, since the computational burden grows slower than in MILP-based approaches. Both the cost saving potential due to an increased flexibility and the reduced computational burden are shown in preliminary studies in Section VI.

Notation: Throughout the paper, $k \in \{1,\ldots,\mathscr{K}\}$ denotes the time slots where the k^{th} slot is defined as the interval $[t_k, t_{k+1}]$ of length $\Delta t_k = t_{k+1} - t_k$, the index $i \in \{1,\ldots,N\}$ refers to a smart appliance, and the index $j \in \{1,\ldots,M_i\}$ refers to its corresponding work phase. The total amount of power that has to be provided to the individual phases is denoted by $x^{\Sigma}_{i,j}$ and the power consumption of an appliance by p_i. Binary variables or functions are denoted by $\alpha(k)$ or $\alpha(t)$, respectively.

II. Modeling of Devices

This section is concerned with the modeling of the considered devices and storage units (cf. Fig. 1) forming the basis for the derivation of the optimization problem in Section III.

A. Smart Appliances

Each appliance $i \in \{1,\ldots,N\}$ has to go through a number of phases $j \in \{1,\ldots,M_i\}$ in order to finish its respective task. These phases have to be executed in a predefined and fixed order (for example, a washing machine has to heat the water before it soaks the clothes and so forth; cf. [4], [10], [14]). However, the execution of the individual phases comes with some flexibility in its power consumption (for example, the heating of water to a desired temperature can be done very fast, using a lot of energy at once; or slowly, using a smaller amount of energy for a prolonged amount of time). It is assumed that appliances can only enter a finite number of power consumption levels $\mathscr{P}_i = \{p_i^1,\ldots,p_i^{L_i}\}$ with $0 < p_i^1 < \cdots < p_i^{L_i}$ and that the total consumed power is constant. L_i indicates the highest power consumption level an appliance can, or has to, enter – for a washing machine this could for example be during the spin cycle.

Let the remaining power that has to be provided to phase j of appliance i be denoted by $x_{i,j}(t)$. The power consumption of a single phase with a total demand of $x^{\Sigma}_{i,j}$ can then be modeled as

$$\begin{aligned} x_{i,j}(t) &= -\int_{t^0_{i,j}}^{t} \sum_{q=1}^{L_i} \alpha^q_{i,j}(t) p^q_i \,\mathrm{d}t, \\ x_{i,j}\left(t \le t^0_{i,j}\right) &= x^{\Sigma}_{i,j},\; x_{i,j}\left(t \ge t^f_{i,j}\right) = 0, \end{aligned} \tag{1}$$

where $\alpha^q_{i,j}(t) \in \{0,1\}$ are binary indicator functions defining which power consumption level $p^q_i \in \mathscr{P}_i$ is active at time t. Furthermore, $t^0_{i,j}$ and $t^f_{i,j}$ are the beginning and end time of the phase, respectively, which are constraints to its operation. Since phases of the same appliance have to be executed consecutively, these start and end times have to fulfill

$$0 \le t^0_{i,1} \le t^f_{i,1} \le t^0_{i,2} \le \cdots \le t^f_{i,M_i} \le \mathscr{T},\ \forall i \tag{2}$$

and

$$\underline{T}_{i,j} \le t^f_{i,j} - t^0_{i,j} \le \overline{T}_{i,j}, \tag{3}$$

$$t^0_{i,j+1} - t^f_{i,j} \le \Delta_{i,j|j+1}, \tag{4}$$

where $\underline{T}_{i,j}$ and $\overline{T}_{i,j}$ are the minimum and maximum execution times for a phase and $\Delta_{i,j|j+1}$ is the maximum delay between two consecutive phases. The values of these constraints are usually provided by the manufacturer of the appliance. Note that appliances with periodic phases (e.g. refrigerators) can be included straightforwardly by introducing an additional lower bound in equation (4).

Lastly, the indicator functions $\alpha^q_{i,j}(t)$ have to fulfill

$$\sum_{j=1}^{M_i} \sum_{q=1}^{L_i} \alpha^q_{i,j}(t) \le 1 \tag{5}$$

since only one power consumption level can be active at a time.

B. Storages

In addition to smart appliances it is assumed that there are $b = \{1,\ldots,B\}$ energy storage devices (e.g. batteries) available. The state of charge (SOC) $s_b(t)$ during a time interval $\left[t^0_b, t^f_b\right]$ is modeled as

$$\begin{aligned} s_b(t) &= \int_{t^0_b}^{t} \left(\alpha^c_b(t) p^c_b - \alpha^d_b(t) p^d_b\right) \mathrm{d}t, \\ \underline{s}_b &\le s_b(t) \le \overline{s}_b \end{aligned} \tag{6}$$

where $\underline{s}_b, \overline{s}_b$ are lower and upper bounds on the SOC, and $\alpha^c_b(t) \in \{0,1\}, \alpha^d_b(t) \in \{0,1\}$ are once again binary functions that indicate if the battery is charging – consuming a power of p^c_b – or discharging – providing a power of p^d_b to the grid. Since it is assumed here that the battery can only be in one of the two states, α^c_b and α^d_b have to fulfill

$$\alpha^c_b(t) + \alpha^d_b(t) \le 1\,. \tag{7}$$

Smart appliances and energy storages together form a nanogrid, as is exemplarily depicted in Fig. 1.

III. Formulation of the Optimization Problem

The goal of this work is to minimize the power bill for appliance operation. Since some of the required power can be provided by the storage units, the total power exchanged with the external grid during a time window $[0,\tau]$ is given by

$$P_{ex}(t) = \sum_{i=1}^{N} \sum_{j=1}^{M_i} \sum_{q=1}^{L_i} \alpha^q_{i,j}(t) p^q_i + \sum_{b=1}^{B} \left(\alpha^c_b(t) p^c_b - \alpha^d_b(t) p^d_b\right). \tag{8}$$

Using (8), the value function to be minimized on the time horizon $[0,\mathscr{T}]$ is given by

$$\mathscr{J}\left(\alpha^q_{i,j}, \alpha^c_b, \alpha^d_b, t\right) = \int_0^{\mathscr{T}} E(t) P_{ex}(t)\,\mathrm{d}t \tag{9}$$

Here, $E(t)$ is the power price profile. Since this profile can generally be assumed to be piecewise constant (e.g. over periods of 15 minutes in Europe's day ahead market), the time interval $[0, \mathscr{T}]$ can be divided into $\mathscr{K}$ time slots $k \in \{1, \dots, \mathscr{K}\}$ and the general optimal scheduling problem to be solved is given by

$$\begin{aligned} \min_{\substack{\alpha_{i,j}^q, \alpha_b^c, \alpha_b^d, \\ t_{i,j}^0, t_{i,j}^f}} \; & \sum_{k=1}^{\mathscr{K}} \left(E(k) \int_{t_k}^{t_{k+1}} P_{ex}(t) \, \mathrm{d}t \right) \\ \text{s.t. } & (1), (2), (3), (4), (5), (6), (7), (8), \\ & \alpha_{i,j}^q(t), \alpha_b^c(t), \alpha_b^d(t) \in \{0, 1\}. \end{aligned} \tag{10}$$

However, as defined above, $\alpha_{i,j}^q(t)$, $\alpha_b^c(t)$, and $\alpha_b^d(t)$ are continuous time binary functions so the problem is still infinite-dimensional. A straight-forward approach would be to define binary variables $\alpha_{i,j}^q(k)$ that indicate the power consumption level for a whole time slot k. The constraints (2)-(4) on the beginning and end times of the individual phases could then be enforced via auxiliary binary variables [14], [15].

Remark 1: The solution to problem (10) is not unique for an obvious reason: Since the binary functions $\alpha(t)$ are time-continuous but the power price is piecewise constant, the two signals

$$\alpha_1(t) = \begin{cases} 1 & 0 \le t \le 0.5 \\ 0 & 0.5 < t \le 1 \end{cases}$$

and $\alpha_2(t) = 1 - \alpha_1(t)$, assuming a constant power price E_0 on the interval $[0, 1]$, would lead to the same cost since

$$\int_0^1 \alpha_1(t) E_0 \, \mathrm{d}t = \int_0^{0.5} E_0 \, \mathrm{d}t = 0.5 E_0 = \int_{0.5}^1 E_0 \, \mathrm{d}t = \int_0^1 \alpha_2(t) E_0 \, \mathrm{d}t$$

The following chapter shows how problem (10), which usually scales badly with the problem size, can be solved without using any binary variables exploiting the non-uniqueness of its solution.

IV. Integral Reformulation

Instead of using the binary indicator functions $\alpha_{i,j}^q(t)$, the normalized values of their integrals

$$w_{i,j}^q(k) := \frac{1}{t_{k+1} - t_k} \int_{t_k}^{t_{k+1}} \alpha_{i,j}^q(t) \, \mathrm{d}t \tag{11}$$

are introduced as new decision variables, which amount to the fraction of a (fixed) time slot k a phase is supplied with power level p_i^q. Since the $\alpha_{i,j}^q(t)$ can take the values 0 or 1 for arbitrary amounts of time within a given time interval (cf. Remark 1) it follows that $w_{i,j}^q(k) \in [0, 1]$. This approach is sometimes also referred to as *workload partitioning* [17], [18]. Subsequently, equation (5) has to be reformulated to match the newly introduced decision variables and some additional constraints are introduced in order to ensure the desired operation:

$$w_{i,j}^q(k) = 0, \; \forall k : \left\{ t_{k+1} < t_{i,j}^0, t_k \ge t_{i,j}^f \right\}, \tag{12a}$$

$$\sum_{q=1}^{L_i} w_{i,j}^q(k) = 1, \; \forall k : \left\{ t_k \ge t_{i,j}^0, t_{k+1} \le t_{i,j}^f \right\}, \tag{12b}$$

$$\sum_{q=1}^{L_i} \Delta t_k w_{i,j}^q(k) = t_{k+1} - t_{i,j}^0, \; \forall k : \left\{ t_{i,j}^0 \in [t_k, t_{k+1}] \right\}, \tag{12c}$$

$$\sum_{q=1}^{L_i} \Delta t_k w_{i,j}^q(k) = t_{i,j}^f - t_k, \; \forall k : \left\{ t_{i,j}^f \in [t_k, t_{k+1}] \right\}. \tag{12d}$$

Equation (12a) ensures that there is no power provided to the appliance outside of its specified start and end times, while (12b) prevents interrupts during the execution of a phase. Equations (12c) and (12d) guarantee that the phases start and end at their specified times.

Furthermore, the model of the appliances (1) can be rewritten as the difference equation

$$\begin{aligned} x_{i,j}(k+1) &= x_{i,j}(k) - \sum_{q=1}^{L_i} \Delta t_k w_{i,j}^q(k) p_i^q \\ x_{i,j}(k \le t_{i,j}^0) &= x_{i,j}^{\Sigma}, \; x_{i,j}(k \ge t_{i,j}^f) = 0, \end{aligned} \tag{13}$$

which has to hold $\forall k \in \left[t_{i,j}^0, t_{i,j}^f \right]$. Again, $x_{i,j}(k)$ is the remaining energy to provide to an appliance i during phase j. The equivalence of (1) and (13) follows directly from linearity.

The same idea can be applied to the storage devices, where the indicator functions $\alpha_b^c(t)$ and $\alpha_b^d(t)$ are replaced by real valued variables $w_b^c(k), w_b^d(k) \in [0, 1]$. Consequently, the dynamics for the state of charge (6) are rewritten as

$$\begin{aligned} s_b(k+1) &= s_b(k) + \Delta t_k w_b^c(k) p_b^c - \Delta t_k w_b^d(k) p_b^d \\ \underline{s}_b &\le s_b(k) \le \overline{s}_b. \end{aligned} \tag{14}$$

The newly introduced variables w_b^c and w_b^d for the storage devices have to fulfill (cf. (7))

$$w_b^c(k) + w_b^d(k) \le 1, \; \forall k. \tag{15}$$

Putting all of this together, the power bought from the external grid during time slot k is

$$\begin{aligned} P_{ex}(k) = & \sum_{i=1}^{N} \sum_{j=1}^{M_i} \sum_{q=1}^{L_i} \Delta t_k w_{i,j}^q(k) p_i^q \\ & + \sum_{b=1}^{B} \Delta t_k \left(w_b^c(k) p_b^c - w_b^d(k) p_b^d \right). \end{aligned} \tag{16}$$

Using equations (13)-(16), the optimization problem to be solved is given by

$$\begin{aligned} \min_{\substack{w_{i,j}^q, w_b^c, w_b^d, \\ t_{i,j}^0, t_{i,j}^f}} \; & \sum_{k=1}^{\mathscr{K}} P_{ex}(k) E(k) \\ \text{s.t. } & (2), (3), (4), (12), (13), (14), (15), (16), \\ & w_{i,j}^q(k), w_b^c(k), w_b^d(k) \in [0, 1]. \end{aligned} \tag{17}$$

In Section V-C, the minimizers $w_{i,j}^{q\star}$, $w_b^{c\star}$, and $w_b^{d\star}$ of problem (17) are then transformed into a working schedule (i.e. a solution to (10)) using McNaughton's wrap around algorithm [18].

Due to equations (12c) and (12d) problem (17) involves conditional constraints of the decision variables. Equation (12c), for instance, states: If $t_{i,j}^0$ lies in the interval $[t_k, t_{k+1}]$, then the sum of all $w_{i,j}^q(k)$ within this interval has to equal $t_{k+1} - t_{i,j}^0$. Implementing such constraints would again require auxiliary binary variables. Hence, the optimization is split into two subproblems, as shown in Section V. Note, however, that so far problem (17) is not a relaxation of problem (10) but provides an exact solution, as also shown in Section V-C.

V. Binary-free Solution Approach

In order to circumvent the usage of binary variables in the implementation of problem (17) it is split into two consecutive optimizations followed by a task ordering step, that derives the actual schedule.

A. Pre-Scheduling

The first subproblem formulates a pre-scheduling of the phases purely based on the constraints on the execution times (2), (3), and (4). The reasoning is, that the minimum and maximum execution times $\underline{T}_{i,j}$ and $\overline{T}_{i,j}$ combined with the total power demand $x_{i,j}^{\Sigma}$ of each phase give a lower and upper bound for the consumed power in each time slot k. Since it is furthermore assumed that $t_{i,j}^0, t_{i,j}^f \in \mathbb{R}_+$, a continuous formulation of the pre-scheduling optimization is desirable. Therefore, a smooth approximation (here an H^{th} order polynomial) of the piecewise constant power price profile $E(k)$ is calculated as

$$\tilde{E}(t) = \sum_{h=0}^{H} \gamma_h t^h \tag{18}$$

with

$$\gamma_h = \arg\min_{\gamma_h} \sum_{k=1}^{\mathscr{K}} \left(\tilde{E}(t_k) - E(k)\right)^2 \tag{19}$$

subject to (18). The pre-scheduling optimization problem is then given by

$$\begin{aligned} \min_{t_{i,j}^0, t_{i,j}^f} \; & \sum_{i=1}^{N} \sum_{j=1}^{M_i} \int_{t_{i,j}^0}^{t_{i,j}^f} \tilde{E}(t)\,\mathrm{d}t \\ \text{s.t. } & (2), (3), (4). \end{aligned} \tag{20}$$

The minimizers $t_{i,j}^{0\star}, t_{i,j}^{f\star}$ can then be used in equations (12) in the minimization of the phase execution costs.

Remark 2: If the approximated power price profile $\tilde{E}(t)$ is convex, solving problem (20) is straight-forward. This can, for instance, be achieved by choosing the scheduling horizon accordingly, as shown in Section VI. It is, nonetheless, only an approximation of problem (17); the implications of which are discussed in the following subsection.

B. Workload Partitioning

Given the optimal start and end times of the phases based on their respective time constraints, the actual energy consumption during these time windows is now to be optimized. Note, however, that solving the prescheduling (20) separately will in general yield a different (suboptimal) solution than solving (17) directly. One way to mitigate this issue is to interpret $t_{i,j}^{0\star}$ and $t_{i,j}^{f\star}$ as a rough schedule and relax constraints (12c) and (12d) to

$$\sum_{k=1}^{\mathscr{K}} \sum_{q=1}^{L_i} \Delta t_k w_{i,j}^q(k) \le \overline{T}_{i,j} - \underline{T}_{i,j}\,. \tag{21}$$

The main difference between these two equations is that in (21) the optimizer gains some additional flexibility within the first and last time slots of the execution as long as the constraints on the total execution time (cf. equation (3)) are not violated.

The resulting workload partitioning optimization problem is then given by

$$\begin{aligned} \min_{w_{i,j}^q, w_b^c, w_b^d} \; & \sum_{k=1}^{\mathscr{K}} P_{ex}(k) E(k) \\ \text{s.t. } & (12a), (12b), (14), (15), (16), (21) \\ & w_{i,j}^q(k), w_b^c(k), w_b^d(k) \in [0,1]. \end{aligned} \tag{22}$$

It is important to note that during this step the actual price profile $E(t)$ (not its approximation) is used.

In the following it is shown how the minimizers $w_{i,j}^{q\star}$, $w_b^{c\star}$, and $w_b^{d\star}$ can be used to create a solution to problem (10).

C. Task Ordering

In order to produce a working schedule from the minimizers of (22), a variant of McNaughtons wrap-around algorithm is employed [9], [12], [18]. Its main idea is to utilize the fact that the solution to the time-continuous scheduling problem (10) is not unique by choosing a specific solution that fulfills the constraints based on simple rules. It takes the minimizers $w_{i,j}^{q\star}$, $w_b^{c\star}$, and $w_b^{d\star}$ as inputs and outputs the start and end times $\sigma_{i,j}^q(k)$ and $\eta_{i,j}^q(k)$ of a phase (or more specifically: the start and end times of the periods, phase (i,j) is supplied with power level q), which are the time points at which the corresponding binary indicator function $\alpha_{i,j}^q(t)$ would switch from 0 to 1 (σ) or from 1 to 0 (η), respectively. The detailed implementation is shown in Algorithm 1. Therefore, the optimal binary indicator functions to (10) are given by

$$\alpha_{i,j}^{q\star}(t) = \begin{cases} 1 & t_k + \Delta t_k \sigma_{i,j}^q(k) \le t \le t_k + \Delta t_k \eta_{i,j}^q(k) \\ 0 & \text{else.} \end{cases} \tag{23}$$

Once again, the same procedure can be repeated for the storage devices, yielding $\alpha_b^{c\star}(t)$ and $\alpha_b^{d\star}(t)$.

Remark 3: Note that since only *for*-loops are involved, this algorithm terminates in finite time. Furthermore it executes very fast because only basic and simple operations such as summations and "smaller than" comparisons of real numbers are involved.

Algorithm 1: Task ordering algorithm.

Input : $\{w_{i,j}^{q\star}(k) \in [0,1] \mid j \in J, q \in Q\}$

for $j = 1, \ldots, j_{\max}$ **do**
  for $q = 1, \ldots, q_{\max}$ **do**
    if $j = 1$ *and* $q = 1$ **then**
      $\eta_{1,1}^{q}(k) \leftarrow w_{1,1}^{q}(k)$
    else
      if $q = 1$ **then**
        $q_{\text{last}} \leftarrow q_{\max}$
        $j_{\text{last}} \leftarrow j - 1$
      else
        $q_{\text{last}} \leftarrow q - 1$
        $j_{\text{last}} \leftarrow j$
      end
      $\sigma_{i,j}^{q}(k) \leftarrow \eta_{i,j_{\text{last}}}^{q_{\text{last}}}(k)$
      if $\eta_{i,j_{\text{last}}}^{q_{\text{last}}}(k) + w_{i,j}^{q} \leq 1$ **then**
        $\eta_{i,j}^{q}(k) \leftarrow \eta_{i,j_{\text{last}}}^{q_{\text{last}}}(k) + w_{ij}^{q}(k)$
      else
        $\eta_{i,j}^{q}(k) \leftarrow 1$
      end
    end
  end
end

Output: $\{(\sigma_{i,j}^{q}(k), \eta_{i,j}^{q}(k)) \in [0,1] \times [0,1]\}$

VI. Case Studies

In order to show its viability, the proposed approach is implemented for two exemplary scheduling scenarios and compared to the MILP approach (10) using entire time slots (i.e. $\alpha(k)$ instead of $\alpha(t)$). Both scenarios use the intra-day price of power in Germany on October 5th, 2017 [6] as power price profile, depicted in Fig. 2. The scheduling horizon was chosen to be 8 am to 7 pm which results in 44 time slots at a resolution of 15 minutes.

A. Appliances with Long Phases

The first scenario considers three appliances that have three phases and eight power levels each as well as one battery as energy storage unit. The schedules produced by the binary approach are shown in Figs. 3 and 5. The total calculation time for the binary approach was 96.62 s. The final cost after the optimization was 0.746 €.

The schedules produced by the integral reformulation with 2-step optimization and task ordering as proposed in this paper are shown in Figs. 4 and 6. It took 5.775 s in total to perform the calculation: 0.39 s for the prescheduling optimization, 5.37 s for the optimal workload partitioning, and 0.015 s for the task ordering for all phases and the batteries. The final cost was 0.654 €.

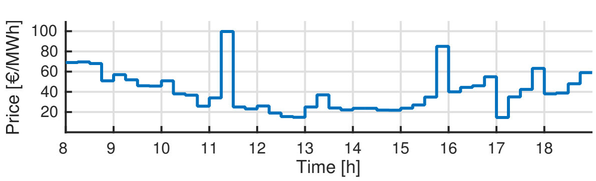

Fig. 2. Power price in Germany on October 15th, 2017 [6]. The price is piecewise constant over intervals of 15 minutes.

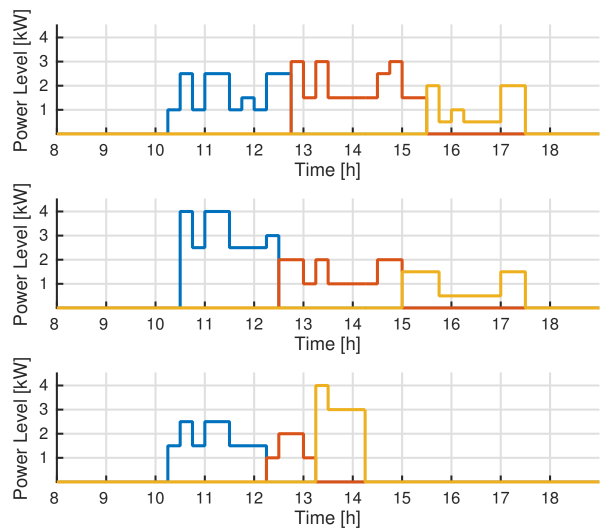

Fig. 3. Schedule for the individual phases of the three appliances, each shown in a separate subfigure, using the binary problem 10 with fixed binaries per time slot and auxiliary variables for constraint satisfaction (cf. [14]). The different colors denote the different phases of each appliance.

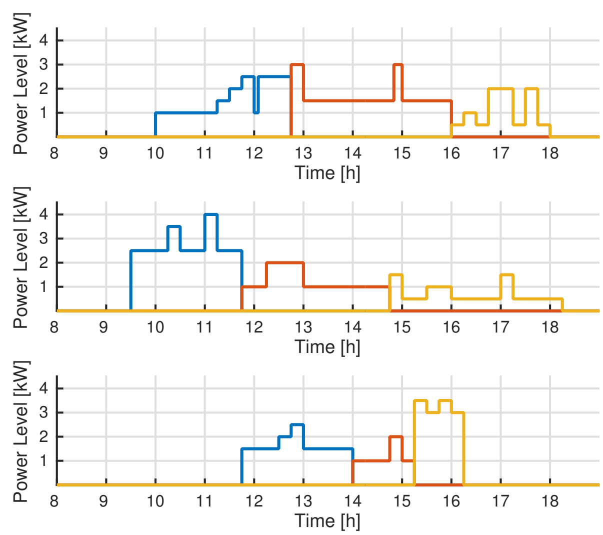

Fig. 4. Schedule for the individual phases of the three appliances, each shown in a separate subfigure, using the binary-free 3-step reformulation. The different colors denote the different phases of each appliance.

B. Appliances with Short Phases

In the second scenario four appliances with five phases and again eight power levels each were scheduled. Due to a lack of space only the numerical results are presented here.

The MILP approach took 1820.58 s and lead to a cost of 0.401 €. In contrast, the presented approach took a total of 35 s: 0.46 s for the prescheduling, 32.56 s for the workload partitioning, and 0.047 s for the task ordering. It was able achieve a cost of 0.299 €.

Note that even a slight increase of the problem size leads to a significantly higher computational effort in both cases. However, while the calculation time of the MILP approach increased by a factor of 19, the time of the presented approach increased only by a factor of 6.

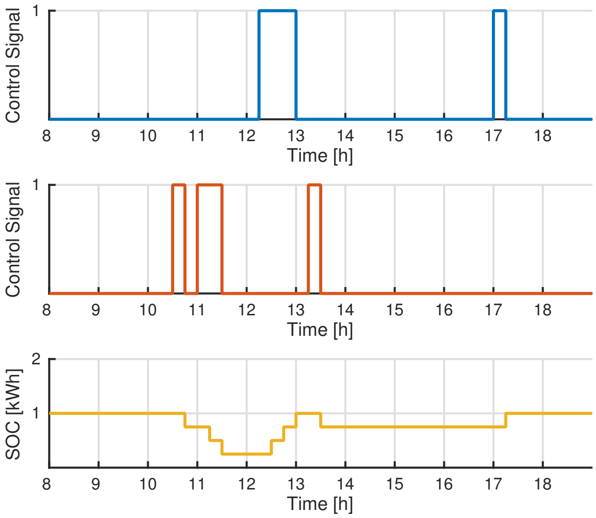

Fig. 5. Schedules for the charging (top) and discharging (middle) of the battery as well as its state of charge (bottom) using the binary problem 10 with fixed binaries per time slot and auxiliary variables for constraint satisfaction (cf. [14]).

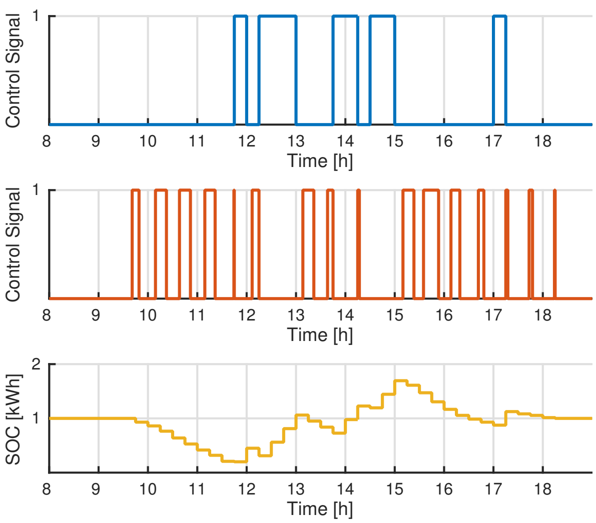

Fig. 6. Schedules for the charging (top) and discharging (middle) of the battery as well as its state of charge (bottom) using the binary-free 2-step reformulation with task ordering.

Remark 4: All computations were performed on a MacBook Pro with an 2.7 GHz Intel Core i5 CPU and 8 GB 1867 MHz DDR3 RAM. All problems were solved using YALMIP and CPLEX in MATLAB 2015a.

VII. Conclusions & Outlook

A binary-free approach for the time-continuous scheduling of smart appliances was presented. It is based on an integral reformulation of a continuous mixed-integer optimization problem and employs a 2-step optimization as well as a task ordering step. Preliminary case studies demonstrated the potential for a better scaling with problem size of the proposed approach than the original MILP based approach.

Since the models for the appliances and storage devices are disregarding internal dynamics so far, an inclusion of such would yield more realistic results. It might also be desirable to constrain the maximum number of charging or discharging cycles in order to extend the battery life, as for example done in [14].

Furthermore, a detailed analysis of the computational complexity for the presented approach would be necessary in order to fully evaluate its potential for future large scale optimal scheduling approaches.

References

[1] M. R. Alam, M. B. I. Reaz, and M. A. M. Ali, “A review of smart homes - past, present, and future,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, vol. 42, pp. 1190–1203, 2012.

[2] C. Chen, S. Kishore, and L. V. Snyder, “An innovative rtp-based residential power scheduling scheme for smart grids,” *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5956–5959, 2011.

[3] C. Chen, J. Wang, Y. Heo, and S. Kishore, “MPC-based appliance scheduling for residential building energy management controller,” *IEEE Transactions on Smart Grid*, vol. 4, pp. 1401–1410, 2013.

[4] K. Cheong Sou, J. Weimer, H. Sandberg, and K. Johansson, “Scheduling smart home appliances using mixed integer linear programming,” in *IEEE Conference on Decision and Control*, 2011, pp. 5144–5149.

[5] P. Du and N. Lu, “Appliance commitment for household load scheduling,” in *IEEE PES Transmission and Distribution*, 2012.

[6] EPEXSPOT, “Continuous intraday market data.” [Online]. Available: https://www.epexspot.com/de/marktdaten/intradaycontinuous/intraday-table/2017-10-05/DE

[7] A. Ghasem Azar and R. H. Jacobsen, “Appliance scheduling optimization for demand response,” *International Journal On Advances in Intelligent Systems*, vol. 9, no. 1&2, pp. 50–64, 2016.

[8] M. He, S. Murugesan, and J. Zhang, “Multiple timescale dispatch and scheduling for stochastic reliability in smart grids with wind generation integration,” in *IEEE INFOCOM*, 2011.

[9] D. Karger, C. Stein, and J. Wein, “Scheduling algorithms,” in *Algorithms and Theory of Computation Handbook*, M. J. Atallah and M. Blanton, Eds. Chapman & Hall/CRC, 2010, ch. Scheduling Algorithms, pp. 1–34.

[10] K. S. Kim, S. Lee, T. O. Ting, and X.-S. Yang, “Atomic scheduling of appliance energy consumption in residential smart grid,” *Computing Research Repository*, 2015.

[11] J. Ma, H. H. Chen, L. Song, and Y. Li, “Residential load scheduling in smart grid: A cost efficiency perspective,” *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 771–784, 2016.

[12] R. McNaughton, “Scheduling with deadlines and loss functions,” *Management Science*, vol. 6, no. 1, pp. 1–12, 1959.

[13] B. Nordman, K. Christensen, and A. Meier, “Think globally, distribute power locally: The promise of nanogrids,” *Computer*, vol. 45, no. 9, pp. 89–91, Sept 2012.

[14] K. Paridari, A. Parisio, H. Sandberg, and K. H. Johansson, “Robust scheduling of smart appliances in active apartments with user behavior uncertainty,” *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 1, pp. 247–259, 2016.

[15] I. Prodan, E. Zio, and F. Stoican, “Fault tolerant predictive control design for reliable microgrid energy management under uncertainties,” *Energy*, vol. 91, pp. 20–34, 2015.

[16] H. M. Rad, V. W. S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, “Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid,” *IEEE Transactions on Smart Grid*, vol. 1, pp. 320–331, 2010.

[17] J. Shen, A. L. Varbanescu, Y. Lu, P. Zou, and H. Sips, “Workload partitioning for accelerating applications on heterogeneous platforms,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 9, pp. 2766–2780, 2016.

[18] M. Thammawichai and E. C. Kerrigan, “Feedback scheduling for energy-efficient real-time homogeneous multiprocessor systems,” in *IEEE Conference on Decision and Control*, 2016, pp. 1643–1648.

[19] M. van den Briel, P. Scott, and S. Thiébaux, “Randomized load control: A simple distributed approach for scheduling smart appliances,” in *International Joint Conference on Artificial Intelligence*, 2013.

[20] P. Yi, X. Dong, A. Iwayemi, C. Zhou, and S. Li, “Real-time opportunistic scheduling for residential demand response,” *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 227–234, 2013.