

# A Novel Smart Home Energy Management System: Cooperative Neighbourhood and Adaptive Renewable Energy Usage

Matteo Cabras, Virginia Pilloni, Luigi Atzori  
DIEE, University of Cagliari, Italy  
{virginia.pilloni,l.atzori}@diee.unica.it

**Abstract**—Energy usage optimization in Smart Homes is a critical problem: over 30% of the energy consumption of the world resides in the residential sector. Usage awareness and manual appliance control alone are able to reduce consumption by 15%. This result could be improved if appliance control is automatic, especially if renewable sources are present locally. In this paper, a Smart Home Energy Management system that aims at automatically controlling appliances in groups of smart homes belonging to the same neighborhood is proposed. Not only is electric power distribution considered, but also renewable energy sources such as wind micro-turbines and solar panels. The proposed strategy relies on two algorithms. The Cost Saving Task Scheduling algorithm is aimed at scheduling high-power controllable loads during off-peak hours, taking into account the expected usage of the non-controllable appliances such as fridge, oven, etc. This algorithm is run whenever a new need of energy from a controllable load is detected. The Renewable Source Power Allocation algorithm re-allocated the starting time of controllable loads whenever surplus of renewable source power is detected making use of a distributed max-consensus negotiation. Performance evaluation of the algorithms tested proves that the proposed approach provides an energy cost saving that goes between 35% and 65% with reference to the case where no automatic control is used.

**Index Terms**—Renewable sources; Smart Home; Energy Management Systems

## I. INTRODUCTION

The last few years have been characterized by the technological revolution of the Internet of Things (IoT) [1]. The aim of this paradigm is to enable the network objects to dynamically cooperate and make their resources available, in order to reach a common goal, i.e. the reduction of energetic consumption in a building. One of the main application areas related to IoT is represented by Smart Homes, and particularly Smart Home Energy Management (SHEM) systems [2]. Smart Homes are residential buildings equipped with devices which cooperate in order to achieve a common set of goals. Some key features characterize many Smart Home environments: i) available node energy is often limited. This is the case, for example, of battery powered nodes, which have limited energy amounts. ii) Smart devices, which give the opportunity to monitor and to remotely control key equipment within homes. iii) Decision-support tools aimed to aid users in making more intelligent

decisions and based on maximizing the benefits gained by the end users when they utilize energy services.

The importance of energy usage optimization in Smart Homes is proven by statistics, which indicates that the electricity consumption in the residential sector represents over 30% of the energy consumption of the world [3]. As demonstrated by the literature, usage awareness alone has the potential to reduce consumption by 15% in private households [4]. In particular, the effects of the Italian Time-of-Use (TOU) tariffs-based Demand Side Management (DSM) program on demand and load shifting were examined in [5]. Comparing results with flat tariffs, it was observed that TOU tariffs lead to higher electricity demand and lower prices values.

The problem of SHEM is treated in many different studies: [4] and [6] propose a middleware for energy awareness integration into Smart Homes; [7] studies an automatic cost-effective light adjustment system; [8] and [9] introduce SHEM systems that take into account Renewable Energy Sources (RES). None of the analysed studies focus on automatic scheduling and control mechanisms of controllable appliances based on: TOU tariffs, RES power, and a *User Profile* inferred through a predictive model by appliance usage.

In this work we consider a Smart Home scenario where the aim is to reduce the electricity cost by monitoring energy consumption habits and RES production, and dynamically shifting tasks of controllable appliances. The approach is twofold:

- the Cost Saving Task Scheduling (CSTS) algorithm schedules tasks characterised by high power load in off-peak times, considering the User Profile. The task starting time is postponed as much as possible in order for appliances to wait for available RES power, and consequently cut electrical costs;
- the Renewable Source Power Allocation (RSPA) algorithm uses a max-consensus negotiation among appliances to dynamically choose which tasks should start immediately in order to maximise the use of RES power that is made available by neighbours.

The remainder of the paper is organised as follows. In Section II some past studies and how they approached the SHEM problem are analysed. In Section III, the reference architecture is introduced. Section IV describes the task scheduling model and the algorithms used, specifically CSTS and RSPA. Finally,

in Section V a performance analysis of the proposed algorithms is provided, and in Section VI conclusions are drawn.

## II. PRELIMINARIES

SHEM systems are based on monitoring and controlling household appliances so that their usage is adjusted in a cost-effective way.

Many of the studies proposed in the literature focus on non-automated SHEM systems, in which suitable advices are provided to the users. Jahn et al. [4] propose a middleware for energy awareness integration into Smart Homes. By using this middleware, users can observe their appliances' energy consumption and make decisions to reduce energy costs. In [6] an optimization framework where the scheduling time of appliances is suggested is described. The aim of this work is to shift appliances' load to off-peak time, in order to reduce electricity costs.

In the last years SHEM systems have evolved into more dynamic systems, where controllable appliances are monitored and managed by a central controller. A cost-effective control system for the reduction of lighting energy consumption is provided in [7]. The authors of this paper studied how home energy consumption can be improved by automatically adjusting lights based on room occupancy, daylight and time of the day. In [8] a SHEM system where appliances are controlled taking into account renewable energy gathered from the houses within the same neighbourhood is described. RES are considered also in [9], which further makes use of predictive models for short term power forecasts of the RES. Accordingly, the authors address the problem of preventing the challenges due to the sporadic nature of wind and solar power generation in designing scheduling techniques.

Although many studies on appliances usage profiles and prediction have been accomplished in [10][11][12][13], there are still no studies where predictive models on appliance usage and RES are used in automatised SHEM systems.

## III. REFERENCE SCENARIO

In this work we consider a Smart Home scenario where the aim is to dynamically postpone or bring forward the execution of tasks of controllable appliances so that the electricity costs are reduced. We refer to controllable appliances as to those whose start can be delayed, provided that they are executed before a given deadline. Our reference scenario is that of a group of houses such as a block or a condominium, which we call Cooperative Neighbourhood.

Inside each house there are appliances (e.g. electric oven, fridge, boiler, battery charger, light bulbs) that consume energy. On the other hand, power supplies such as electric grid, solar panels, and micro wind turbine provide energy that can be used to run appliances. Smart Meters and actuators are associated to these appliances to monitor their energy consumption/production and control their activation/deactivation.

The appliances are divided into 4 groups, based on their characteristics and requirements:

- Group 1: small loads such as lights, battery chargers;
- Group 2: not controllable high loads such as ovens, heaters;

Group 3: controllable loads such as washing machines, dryers, electric cars;

Group 4: supplies such as solar panels, micro wind turbines.

One or more *tasks* are associated to each consuming appliance. We refer to the set  $k = \{1, \dots, K\}$  of tasks associated to an appliance as to the functions that it is able to carry out. For instance, the set of tasks of a washing machine consists of all its washing cycles. A particular case is represented by appliances that are able to perform only one task, such as the TV, which can only be turned on. For these cases, the set of tasks is made of one single element. A power consumption amount is associated to each task. At first, information related to involved appliances' characteristics, and tasks that they are able to perform, will be detected by Smart Meters and sent to a *Central Unit*. Users' habits, i.e. how family members usually use appliances, are monitored and sent to the Central Unit as well. Based on this information, a profile of their energy consumption habits, namely User Profile, will be associated to users. If, for example, the house is empty during working hours, it is unlikely that appliances such as TV or lights are turned on during this span of time. At a later stage, information acquired and processed by the Central Unit is delivered to the appropriate Virtual Objects (VOs). As depicted in Figure 1, each VO is responsible for managing the communication of all the appliances inside a house. More precisely, each VO acts as an interface between the appliances of a single house and the central unit. The role of the VO can be taken by any Smart Meter that monitors the house appliances.

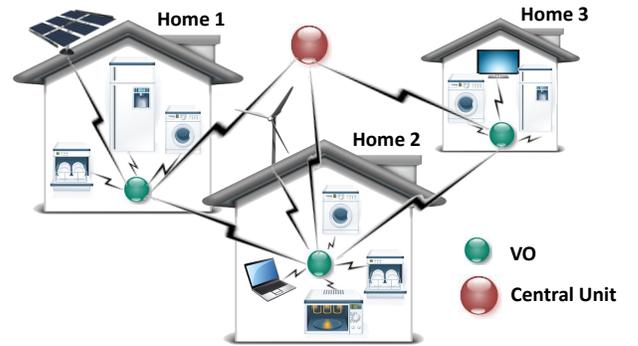


Fig. 1. Reference scenario

We index the appliances with  $i \in \{1, \dots, I\}$  and the homes with  $h \in \{1, \dots, H\}$ . Each house's VO, namely  $VO_h$ , stores the following information about appliance  $i$ , depending on which Group it belongs to:

- Group 1: condition  $i \in G_h^1$ , where  $G_h^1$  is the set of appliances of Group 1 inside home  $h$ ; state (on/off)  $x_{ik}(t)$  for appliance  $i$  related to task  $k$ , at time  $t$  (often they are able to perform only one task); power  $P_{ik}^{cons}$  consumed by appliance  $i$  to carry on task  $k$ ; probability  $Pr_{ik}(t)$  that appliance  $i$  performs task  $k$  at time  $t$ , as indicated by the User Profile. Since power consumption for appliances of this Group is negligible, we suppose a fixed energy consumption when they are on. Therefore, information about

power consumption is delivered by the Smart Meter only the first time;

Group 2: condition  $i \in G_h^2$ , where  $G_h^2$  is the set of appliances of Group 2 inside home  $h$ ; state (on/off)  $x_{ik}(t)$  for appliance  $i$  related to task  $k$ , at time  $t$ ; power  $P_{ik}^{cons}$  consumed by appliance  $i$  to carry on task  $k$ ; probability  $Pr_{ik}(t)$  that appliance  $i$  performs task  $k$  at time  $t$ , as indicated by the User Profile or based on user needs (e.g. if the video recorder is set to turn on at time  $t_{ik}^{ST}$  and turn off at time  $t_{ik}^{DL}$ ,  $Pr_{ik}(t_{ik}^{ST} \leq t \leq t_{ik}^{DL}) = 1$ );

Group 3: condition  $i \in G_h^3$ , where  $G_h^3$  is the set of appliances of Group 3 inside home  $h$ ; state (on/off)  $x_{ik}(t)$  for appliance  $i$  related to task  $k$ , at time  $t$ ; power  $P_{ik}^{cons}$  consumed by appliance  $i$  to execute task  $k$ ; time  $t_{ik}^{exec}$  needed by appliance  $i$  to perform task  $k$ ; deadline  $t_{ik}^{DL}$  before which appliance  $i$  needs to perform task  $k$ ; time  $t_{ik}^{ST}$  when appliance  $i$  started to perform task  $k$ , if task  $k$  is running (i.e.  $x_{ik}(t) = ON$ );

Group 4: condition  $i \in G_h^4$ , where  $G_h^4$  is the set of appliances of Group 4 inside home  $h$ ; state (on/off)  $x_{ik}(t)$  for appliance  $i$  at time  $t$ ; power  $P_{ik}^{prod}(t)$  produced by appliance  $i$  at time  $t$ . Note that the supplier can perform only one task, so  $k$  is only equal to 1. Nevertheless, we keep subscript  $k$  for notation convenience.

#### IV. TASK SCHEDULING MODEL

The proposed SHEM system is designed to perform three basic functions:

- It monitors and analyses users' habits with reference to appliance usage. Based on this information, a User Profile is created.
- It detects power surplus due to RES production and distributes this power to the houses of the same neighbourhood, with the aim of maximising its consumption.
- It sets the most convenient starting time of controllable appliances so that their tasks are executed when it is more convenient, according to TOU tariffs and RES energy production. In order to accomplish this function, two algorithms are developed:
  - The CSTS, which schedules tasks characterised by high power load in off-peak times, considering the User Profile;
  - The RSPA, which dynamically shifts tasks in order to maximise the use of renewable energy that is made available by neighbours.

The sequence of steps to be performed is shown in Figure 2. As soon as appliance  $i$  placed in home  $h$  needs to start task  $k$ , it sends an activation request to  $VO_h$ . If appliance  $i$  is not controllable or it is not a supplier (i.e. it belongs to  $G_h^1$  or  $G_h^2$ ) it just needs to notify to  $VO_h$  that it is changing state ( $x_{ik}(t) = ON$ ) for the whole duration of the task.  $VO_h$  sets its probability to be on to 1 accordingly. When task  $k$  stops, appliance  $i$  informs  $VO_h$ , which sets  $Pr_{ik}(t)$  to its probability to turn on again, according to the User Profile. Its power

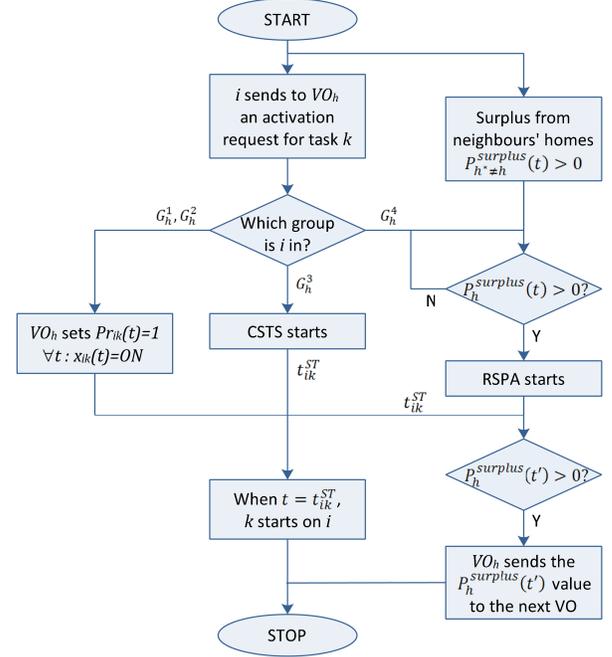


Fig. 2. Task assignment steps

consumption and duration values are monitored and sent to the Central Unit, which analyses them and updates the User Profile accordingly.

If appliance  $i$  is a controllable consumer, i.e. it belongs to  $G_h^3$ , CSTS is started. CSTS is a centralized algorithm that is performed by the VO to postpone the starting time  $t_{ik}^{ST}$  of  $G_h^3$  appliances, so that their tasks are executed during off-peak hours, when electricity charge is lower. The user can set the minimum starting time  $t_{ik}^{minST}$  and the deadline  $t_{ik}^{DL}$  when the task needs to be carried out. Therefore, the starting time  $t_{ik}^{ST}$  is computed by the CSTS according to the user preferences, provided that the available power  $P^{max}$  is not exceeded by the simultaneous usage of the appliances that made an activation request.

If appliance  $i$  is a supplier (i.e. it belongs to  $G_h^4$ ), or a power surplus coming from neighbouring houses is detected by the  $VO_h$ , it computes the  $P_h^{surplus}(t)$  value of the power surplus related to house  $h$  at time  $t$ .  $P_h^{surplus}(t)$  takes into account all the power surplus contributions that are made available by the neighbour houses along with the power supplied by  $G_h^4$  appliances, and it is decreased by the power consumed by the appliances inside home  $h$  if they are on

$$\begin{aligned}
 P_h^{surplus}(t) = & \sum_{h^* \neq h} P_{h^*}^{surplus}(t) - \sum_{i \in \{G_h^1, G_h^2\}, k} P_{ik}^{cons} \cdot Pr_{ik}(t) \\
 & - \sum_{i \in G_h^3, k} P_{ik}^{cons} \cdot x_{ik}(t) + \sum_{i \in G_h^4, k} P_{ik}^{prod}(t)
 \end{aligned} \tag{1}$$

Whenever  $P_h^{surplus}(t) > 0$  is verified,  $VO_h$  broadcasts this information to the appliances it controls.

If there is any  $G_h^3$  appliance that is waiting to turn on and its power consumption is lower than the available surplus power,

RSPA is started. RSPA is a distributed consensus algorithm where appliances compete for the same resource, negotiating among each other. After the algorithm has converged, those appliances that have won the negotiation immediately turn on. If there is any surplus power still available, it is sent to the closest VO.

#### A. Cost Saving Task Scheduling algorithm

The CSTS is a centralized algorithm based on the concept that, whenever possible, tasks that can be postponed should be performed during off-peak hours, when electricity charge is lower.

When appliance  $i \in G_h^3$  sends to  $VO_h$  an activation request, it sends its deadline value  $t_{ik}^{DL}$  and its minimum starting time  $t_{ik}^{minST}$ . Consequently,  $VO_h$  starts CSTS to assign/reassign to all  $G_h^3$  appliances the most convenient starting time according to TOU tariffs. Hence, a suitable starting time  $t_{ik}^{ST}$  in the range  $[t_{ik}^{minST}, t_{ik}^{DL} - t_{ik}^{exec}]$  is computed, provided that the available power  $P^{max}$  is not exceeded by the simultaneous usage of several appliances. The optimization only takes into account consumer appliances and their probability to be turned on. It neglects suppliers, whose power is negotiated among appliances during RSPA. Note that it is preferable that appliances wait for available RES power as long as it is possible, so that electrical costs are cut. For this reason, CSTS assigns the most convenient  $t_{ik}^{ST}$  that is closest to  $t_{ik}^{DL}$ .

Finding an optimal scheduling assignment is an NP-hard problem [14], which complexity scales exponentially with the problem size. In order to reduce the complexity of the algorithm, and thus its convergence time and energy needed to be run, we propose a greedy approach, which is characterised by a linear complexity. As described in details in Algorithm 1, the concept on the basis of CSTS is that those tasks that consume more energy, i.e. those that present higher values of energy consumption  $E_{ik}^{cons} = P_{ik}^{cons} \cdot t_{ik}^{exec}$ , are those that generate more energy cost saving when they are shifted to off-peak hours. Therefore, those tasks have the priority to be scheduled for those hours where TOU tariffs are lower, provided that  $P^{max}$  is not exceeded.

Let  $\Lambda_h$  be the array of appliances  $i \in G_h^3$  that made an activation request, and  $E^{cons} = (E_{ik}^{cons})$ . We define a tuple  $\Gamma_h = (\Lambda_h, E^{cons})$  of all the appliances that made a request to  $VO_h$  and their related energy consumption. We also define  $P_h^{tot}(t)$  as the expected instant total power that is likely to be consumed at time  $t$  by all non-controllable appliances managed by  $VO_h$  as

$$P_h^{tot}(t) = \sum_{i \in \{G_h^1, G_h^2\}, k} P_{ik}^{cons}(t) \cdot Pr_{ik}(t) \quad (2)$$

$P_h^{tot}(t)$  is updated whenever the probability  $Pr_{ik}(t)$  changes. The sequence of steps of CSTS is described as follows.  $\hat{P}^{tot}(t)$  is initialised with the value of  $P_h^{tot}(t)$ . The tasks of the controllable appliances that made an activation request are then sorted in descending order with respect to their energy consumption value. Starting from the task with the highest  $E^{cons}$ , the starting time to which corresponds the lowest electrical energy cost  $C^{min}$  is found. If there is more than

one  $t_{ik}^{ST}$  that corresponds to the  $C^{min}$  value, the algorithm assigns the highest one. In this way, if some  $P_h^{surplus}(t)$  is available, the task has more probability to be able to negotiate to start before the assigned  $t_{ik}^{ST}$ . The total power consumption is then updated for the time when the task is expected to be in execution.

---

#### Algorithm 1 CSTS

---

```

1: Let  $\hat{P}^{tot}(t) = P^{tot}(t) \forall t$ 
2: Sort in descending order  $\Gamma_h$  by its second element  $E^{cons}$ 
3: for all  $i \in \Gamma_h, k$  do
4:   Let  $C^{min} = \infty$  and  $t_{ik}^{ST} = \infty$ 
5:   for all  $t \in [t_{ik}^{minST}, t_{ik}^{DL} - t_{ik}^{exec}]$  do
6:     if  $\hat{P}^{TOT}(t') + P_{ik}^{cons} \leq P^{max} \forall t' \in [t, t + t_{ik}^{exec}]$ 
7:       then
8:         Let  $C = \sum_{t'=t}^{t+t_{ik}^{exec}} C(t')$ 
9:         if  $C^{min} \geq C$  then
10:            $C^{min} = C$  and  $t_{ik}^{ST} = t$ 
11:         end if
12:       end if
13:   Let  $\hat{P}^{tot}(t) = \hat{P}^{tot}(t) + P_{ik}^{cons} \forall t \in [t_{ik}^{ST}, t_{ik}^{ST} + t_{ik}^{exec}]$ 
14: end for

```

---

#### B. Renewable Source Power Allocation algorithm

Whenever  $VO_h$  detects some surplus power, whether it is caused by RES belonging to home  $h$  or it comes from neighbouring VOs, RSPA is started to distribute this power to the appliances that  $VO_h$  manages. In particular, since  $G_h^1$  and  $G_h^2$  appliances are turned on independently from the VO decisions, RSPA is run to control  $\Gamma_h$  appliances (recall from Section IV-A that  $\Gamma_h$  is the array of controllable appliances that made an activation request to the VO).

Since surplus power value continuously change, the algorithm needs to be as lightweight as possible to quickly adapt to changes. Furthermore, communication with appliances that are not visible from the VO needs to be quick. For these reasons, RSPA is chosen to be a distributed algorithm, where appliances negotiate in order to reach a consensus on which one should turn on first.

As described for Algorithm 1, when referring to energy cost saving, tasks that consume more power  $P_{ik}^{cons}$  have the priority to be scheduled when it is more convenient, i.e. when surplus power is available. Furthermore, the priority needs to be given to tasks characterised by closer deadlines. Calling  $t$  the current time, tasks with closer deadlines have higher values of the ratio  $\frac{t}{t_{ik}^{DL} - t_{ik}^{exec}}$ .

Summarising, if the available surplus power is sufficient, RSPA assigns it to the appliances characterised by higher benefit values, defined as

$$b_{ik}(t) = P_{ik}^{cons} \cdot \frac{t}{t_{ik}^{DL} - t_{ik}^{exec}} \quad (3)$$

In order for appliances to reach a consensus on the appliance with the highest  $b_{ik}(t)$  value, a max consensus algorithm is

used. Specifically, a Random-Broadcast-Max consensus algorithm has been chosen for its fast convergence to the solution in wireless channels [15].

Let  $b^{max}$  be the consensus variable and  $b_{ik}^{max}$  be the local consensus variable. The steps of RSPA are described as follows. If some surplus power is detected, its value is broadcast by the VO to its controlled appliances that made an activation request. The algorithm is started by the VO sending the initial benefit value equal to 0. While there is some surplus power available and there are appliances that can use this power, the negotiation runs. Whenever an appliance receives a message with surplus and benefit values, it evaluates its power consumption and benefit. If its power consumption is higher than the available power, and if its benefit is lower than the maximum benefit value, the appliance updates the local value of the maximum benefit and forwards the received message. Otherwise, it sets the maximum benefit value to its benefit value and broadcast the new  $b^{max}$  value, along with the  $P_h^{surplus}(t)$  value. When the timeout is reached, every appliance checks if its local maximum benefit value corresponds to its benefit value. If it is, this means that it task represents the highest benefit. The available surplus power is updated by subtracting the task power consumption, the maximum benefit value it initialised to 0 again so that a new negotiation can start, and the task is started on the appliance that won the negotiation.

---

#### Algorithm 2 RSPA

---

```

1:  $P_h^{surplus}(t) > 0$  is detected by  $VO_h$ 
2: The  $P_h^{surplus}(t)$  value is broadcast by  $VO_h$ , along with a benefit value  $b^{max} = 0$ 
3: while  $P_h^{surplus}(t) > 0$  or at least one appliance can participate to the negotiation do
4:   for all  $i \in \Gamma_h, k$  do
5:     while  $t < timeout$  do
6:       Let  $b_{ik} = b_{ik}(t)$ 
7:       if  $i$  receives a message then
8:         if  $P_{ik}^{cons} \leq P_h^{surplus}(t)$  and  $b_{ik} > b^{max}$ 
9:           Let  $b_{ik}^{max} = b_{ik}$ 
10:        else
11:          Let  $b_{ik}^{max} = b^{max}$ 
12:        end if
13:        Broadcast  $P_h^{surplus}(t)$  and  $b_{ik}^{max}$  values
14:      end if
15:    end while
16:    if  $b_{ik}^{max} = b_{ik}$  then
17:      Update  $P_h^{surplus}(t) = P_h^{surplus}(t) - P_{ik}^{cons}$  and
18:       $b^{max} = 0$  values and broadcast them
19:      Start task  $k$ 
20:    end if
21:  end for
22: end while

```

---

#### V. PERFORMANCE ANALYSIS

The SHEM system described in this paper has been tested supposing to have houses with random user profiles. With

reference to TOU rates, it has been supposed to use the pricing set by the Italian electricity utility company, ENEL. Appliance operation has been simulated and controlled using Arduino Mega 2560 boards [16] with a XBee DigiMesh 2.4 radio module [17]. Power consumption values of real appliances and power production values of real RES power were considered [10]. Data traffic among appliances has been monitored using the X-CTU software [18].

Tests were run supposing to have up to 3 appliances of group  $G_h^3$  per home, with deadlines and minimum starting times set randomly. Results show the energy cost savings obtained when using the proposed SHEM system, with respect to the case where no SHEM system is used.

Since CSTS is run for all the times between  $t_{ik}^{minST}$  and  $t_{ik}^{DL} - t_{ik}^{exec}$ , results' accuracy and computational complexity depend on how the time is discretised, i.e. they depend on the width of the time slot between one time and the next one. Results are shown in Figure 3 for time slots of 10, 20, 30 and 60 minutes, with different numbers of controlled appliances. Solid lines are used to show results for the case where no RES are installed in the houses (i.e. only CSTS is run). On the other hand, dashed lines correspond to the case where RES are present (i.e. both CSTS and RSPA are run). In particular, the power production of a photovoltaic system has been simulated. The produced power has been varied randomly, up to a highest value that is consistent with the one of a commercial photovoltaic home system.

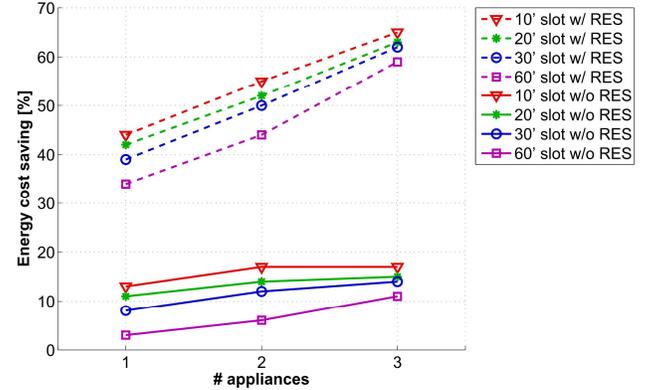


Fig. 3. Energy cost savings for different numbers of controllable appliances and time slot widths. Solid lines correspond to the case with no RES. Dashed line correspond to the case with RES

It is evident that cost saving is much higher when RES are present: it is from 30% to 48% higher with respect to the case without RES. Note that cost savings increase with the number of controlled appliances. However, the slope is steeper when the appliances are fewer, particularly for results with higher accuracy, i.e. 10 minutes time slots. This is because the higher the number of controlled appliances, the lower the power still available, and thus the more difficult the scheduling of all the tasks in off-peak hours. Time slot widths are critical with reference to results' accuracy: a time slot of 10 minutes results in an energy cost saving of about 10% more than that corresponding to a time slot of 60 minutes. This would lead to the conclusion that narrower time slots are

preferable. However, narrower time slots correspond to higher computational complexity.

As demonstrated by Figure 4, CSTS complexity shrinks exponentially with the increment of time slot widths, and complexity for a time slot of less than 10 minutes might become prohibitive for large numbers of controlled appliances. Note that, with a time slot during 20 minutes, complexity decreases by 70%, with a loss in cost saving of just  $2 \div 3\%$ . For this reason, for VOs characterised by low computational complexity, increasing the time slot width represents a good trade-off.

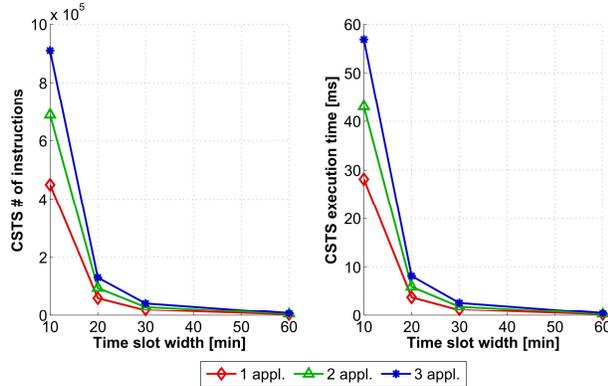


Fig. 4. Complexity of CSTS considering the ATmega2560 microprocessor of the Arduino used in these tests

With reference to the RSPA algorithm, it is a distributed mechanism where each node only make a few comparisons after receiving update messages. For this reason, computational complexity is considered negligible and it is not further analysed.

## VI. CONCLUSIONS

In this paper a SHEM system is proposed. A scheduling model for tasks of controllable appliances that aims to reduce electricity costs is described. In particular, two algorithms are provided: the former, the CSTS, based on the presence of TOU tariffs, shifts the starting time of controllable appliance tasks in off-peak times, taking into account the User Profile, i.e. how the user is expected to use the other appliances in the house. The latter, RSPA, is started whenever a RES installed in a house in the neighbourhood make some power available. In this case, the appliances dynamically negotiate in order to share the available power and start their tasks before the starting time assigned by the CSTS algorithm.

Tests performed using Arduino Mega 2560 boards prove that energy cost saving using the proposed SHEM system goes from 35% to 65%, with reference to the case where tasks are started as soon as they are programmed. Better results are achieved for larger numbers of controlled appliances, although the slope of improvement is steeper for fewer devices.

It is important to note that energy cost savings are strictly related to user habits, with reference to their awareness in the appliance usage. In this sense, SHEM systems are crucial for the automation of this process and to unburden the user from the manual control of controllable appliances.

## REFERENCES

- [1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [2] D.-M. Han and J.-H. Lim, "Smart home energy management system using ieee 802.15. 4 and zigbee," *Consumer Electronics, IEEE Transactions on*, vol. 56, no. 3, pp. 1403–1410, 2010.
- [3] N. Lior, "Sustainable energy development: the present (2009) situation and possible paths to the future," *Energy*, vol. 35, no. 10, pp. 3976–3994, 2010.
- [4] M. Jahn, M. Jentsch, C. R. Prause, F. Pramudianto, A. Al-Akkad, and R. Reiners, "The energy aware smart home," in *Future Information Technology (FutureTech), 2010 5th International Conference on*. IEEE, 2010, pp. 1–8.
- [5] J. Torriti, "Price-based demand side management: Assessing the impacts of time-of-use tariffs on residential electricity demand and peak shifting in northern italy," *Energy*, vol. 44, no. 1, pp. 576–583, 2012.
- [6] F. Baig, A. Mahmood, N. Javaid, S. Razzaq, N. Khan, and Z. Saleem, "Smart home energy management system for monitoring and scheduling of home appliances using zigbee," *J. Basic. Appl. Sci. Res*, vol. 3, no. 5, pp. 880–891, 2013.
- [7] D.-M. Han and J.-H. Lim, "Design and implementation of smart home energy management systems based on zigbee," *Consumer Electronics, IEEE Transactions on*, vol. 56, no. 3, pp. 1417–1425, 2010.
- [8] J. A. Nazabal, F. J. Falcone, C. Fernandez-Valdivielso, and I. R. Matias, "Energy management system proposal for efficient smart homes," in *New Concepts in Smart Cities: Fostering Public and Private Alliances (SmartMILE), 2013 International Conference on*. IEEE, 2013, pp. 1–5.
- [9] A. Tascikaraoglu, A. Boynuegri, and M. Uzunoglu, "A demand side management strategy based on forecasting of residential renewable sources: A smart home system in turkey," *Energy and Buildings*, 2014.
- [10] N. C. Truong, J. McInerney, L. Tran-Thanh, E. Costanza, and S. D. Ramchurn, "Forecasting multi-appliance usage for smart home energy management," in *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, 2013, pp. 2908–2914.
- [11] J. Gil-Quijano and N. Sabouret, "Prediction of humans' activity for learning the behaviors of electrical appliances in an intelligent ambient environment," in *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on*, vol. 2. IEEE, 2010, pp. 283–286.
- [12] A. Kashif, X. H. B. Le, J. Dugdale, and S. Ploix, "Agent based framework to simulate inhabitants' behaviour in domestic settings for energy management," in *ICAART (2)*, 2011, pp. 190–199.
- [13] J. Z. Kolter and J. Ferreira, "A large-scale study on predicting and contextualizing building energy usage," in *AAAI*, 2011.
- [14] V. Pilloni and L. Atzori, "Deployment of distributed applications in wireless sensor networks," *Sensors*, vol. 11, no. 8, pp. 7395–7419, 2011.
- [15] F. Iutzeler, P. Ciblat, J. Jakubowicz *et al.*, "Analysis of max-consensus algorithms in wireless channels," *IEEE Transactions on Signal Processing*, vol. 60, no. 11, pp. 6103–6107, 2012.
- [16] "Arduino mega 2560," <http://arduino.cc/en/Main/ArduinoBoardMega2560>.
- [17] Digi, "Xbee digimesh 2.4," [www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/zigbee-mesh-module/xbee-digimesh-2-4](http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/zigbee-mesh-module/xbee-digimesh-2-4).
- [18] Digi, "X-ctu software," <http://www.digi.com/support/kbase/kbaseresultdet?id=2125>.