HAL
open science

Refined Overlay Power Management in the Home Environment

Han Yan, Bernard Cousin, Cédric Gueguen, Jean-Paul Vuichard

To cite this version:

Han Yan, Bernard Cousin, Cédric Gueguen, Jean-Paul Vuichard. Refined Overlay Power Management in the Home Environment. Greencom, Sep 2014, Taipei, Taiwan. pp.GREENCOM, 10.1109/iThings.2014.94 . hal-01012028

HAL Id: hal-01012028

https://hal.science/hal-01012028

Submitted on 25 Jun 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Refined Overlay Power Management in the Home Environment

Han YAN*‡, Bernard COUSIN*, Cédric GUEGUEN*, Jean-Paul VUICHARD ‡
*University of Rennes1/IRISA; ‡Orange Labs
{han.yan;bernard.cousin;cedric.gueguen}@irisa.fr {han.yan;jeanpaul.vuichard}@orange.com

***Abstract*—The reduction of power consumption plays a key role in numerous environmental and economic issues. Since home network appliances are widely used, residential power consumption makes up a large part of global energy consumption. These home appliances are not only interconnected with each other to provide collaborative services, but are also integrally turned on to contribute to these collaborative services. Faced with this situation, we propose a refined overlay power management system in which appliances can be partially turned on depending on the services, and can be turned on at the moment they are required. In addition, user activities are critical information for the service launch, and so the proposed system has the capacity to learn information about the collaborative service in order to provide efficient power management.**

***Index Terms*—Home network, Energy saving, Green networking, Overlay control network, Low Power**

I. Introduction

In the last decade, many home network devices with exploding power consumption have appeared in our homes. A home network is a complex environment which contains several different types of devices, such as a Set-Top Box (STB), Home GateWay (HGW), workstation, laptop, power line communication plugs and so on, with different kinds of connections, such as Wifi, Ethernet and Powerline communication [1].

Energy saving is recognized as a key issue in global warming and climate change. According to the recent report of the European Commission, Eurostat, there are three dominant energy consumption categories: transport, household and industry. Household energy consumption increased to 26.7% [2] of total energy consumption in 2010 and this category of energy consumption is greater than industry energy consumption. Furthermore, the price of electricity is constantly increasing, with European residential electricity prices increasing on average 2% faster than inflation in 2012. The largest price increases from 2012 to 2013 were observed in Romania (26%) and in Estonia (23%) [3]. It is therefore crucial to reduce energy consumption in home environments.

One challenge is that, in home environments, there has been a proliferation of connected devices and the number of connected devices has led to a sharp increase in energy consumption in the home. With a large number of different kinds of appliances in a home network, such as HGW, STB, network attached storage, computer, etc., household energy saving should take these different connected appliances and their different usages into account.

In the literature, several techniques have been proposed to reduce energy consumption at the device level. Using dynamic power management [4] [5], devices can be switched to a lower power mode when the service demand is reduced. In addition, several algorithms have been proposed to minimize the energy consumption of device components. For instance, Maruti proposed a method that reduces the power supply when there is less traffic over Ethernet links [6], and there are other proposals which aim to control the memory in order to be more power efficient [7] [8]. It is not sufficient, however, to save energy only at the level of each individual device. The power statuses of home devices are interdependent. For example, when all family devices are not operational or not in use, it can be concluded that the home gateway does not need to provide a local network, and its Ethernet and WiFi components can be turned off. The activity or power status of one appliance is not independent information; this information can be used to manage other appliances. Consequently, our solution provides a collaborative system to control the power status of home connected devices at the network level and the power states of functional blocks in these collaborative devices at the device level.

In earlier works, Youn-Kwae Jeong et al. proposed a solution that controls home network devices by reconfiguring the power control element (PCE). Their proposed solution only supplies power to the devices and the functional elements that are related to requested services [9]. In their approach, all functional elements are turned on at the beginning of the service, despite the fact that early functional modules are not needed at that time. The UPnP AV use case [10] can be a good example to explain why there is a time lapse between requested functional blocks in one service: the user controls the home devices with an UPnP Control Point (smart-phone, iPad or other Tablet) and wants to watch a film on his UPnP Media Renderer (STB). This film is saved on his UPnP Media Server (PC). For this service, the content directory functional block on the UPnP Media Server is needed at the beginning of the service. The decoder functional block on the UPnP Media Renderer is required later by the service. Another challenge is that user satisfaction is an important factor in the home network energy control system. There are two important and relevant user parameters: energy cost and waiting delay. In the literature, PCE power management focuses on the energy cost saving solution without considering user satisfaction. In addition to these energy saving management prototypes, there is an increasing interest in taking into the consideration the waiting delay of each solution, because users need to know the

impact of the waiting delay on their service experience before choosing an energy efficient solution. Moreover, different users have different requirements: some prefer to be more energy efficient, some do not want their service experience to be affected, and some want both. Therefore, measuring the delay is important information when proposing a power management solution.

A final challenge is that, in order to be power-efficient and reactive to user demand, the solution should be able to control any device at any time. This requires an always-on connection. In order to ensure power management, the solution proposed by Youn-Kwae Jeong et al. needs a permanent connection (Ethernet or WiFi) which is said to consume more than 1 Watt. Unlike earlier power management solutions which need to maintain a WiFi connection or an Ethernet connection, in our former works [11] [12], we proposed a low power overlay network for the centralized monitoring and control of home network devices by using the following technologies: ZigBee [13], UPnP network [14], 6LowPan [15], and Bluetooth low energy [16]. These can be considered for an infrastructure of a green overlay control network. Home connected appliances can nowadays be turned on by a command from Wakeup on USB [17] or Wake on LAN (WoL) [18]. The main contributions of our refined overlay user-aware efficient power management are: 1) Control of the appliances based on the analysis of the collaborative services which require different functional blocks in different devices. This model helps users to achieve more efficient energy consumption management. 2) Delay measurement based on each service demand. This helps to evaluate the impact of the home network user on the QoS. 3) The uses of an overlay power management network which could be implemented using specific lower power LAN technology, if available.

The rest of the paper is organized as follows. Section II. describes our refined overlay power management. Section III. presents the service pattern, the power consumption model and the delay model. Section IV. presents the setup of the simulation. Section V. analyzes the results of the simulation. Section VI. draws conclusions from our work.

II. The proposed refined overlay energy control power management

In this section, we detail the refined overlay control system which is composed of the power management; the low energy overlay control network; and the refined home network connected devices. Based on this system, we propose two solutions: the Refined Overlay Power Management and the Refined Overlay Auto Learning power management.

A. The refined overlay control system

The proposed system has a power management which controls the functional blocks of devices by sending control messages over a low power overlay network as shown in Fig. 1 (below).

1) Power management: The power management comprises a database and a decision algorithm entity. The database is used to store records of user habits when they use the services. When the service is requested for the first time, the database gathers the information relating to the user request services in order to learn the habits of this family and the information relating to the family user reaction. According to the information that is collected by the system, the power management makes the decisions to control the device with fine granularity. The granularity of the control is said to be fine because the control can turn on/off the functional blocks which are necessary for the collaborative service at the point at which they are requested. We assume each home network collaborative service involves one or more devices which cooperate together in order to meet the service demand of the family.

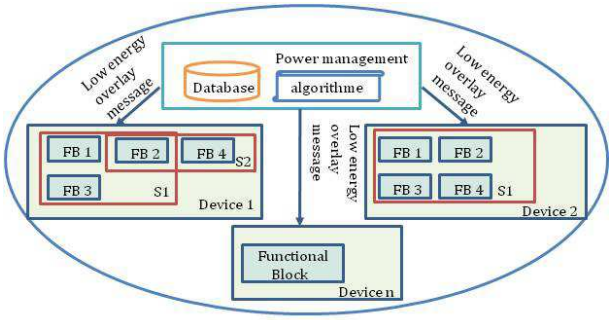


Fig. 1. Home refined network service management

2) Refined network connect devices: In our proposal, we do not only assume that collaborative services involve several devices, but also that each device is refined into one or more functional blocks. At the function level, devices are used as the Functional Blocks (FB) which are needed in different services. In Fig. 1 (above), FB 1, 2, 3 in device 1 and FB 1, 2, 3, 4 in device 2 are requested in service 1. We note that the FB 2 is shared by two services.

Taking the same example of the UPnP AV use case, the user uses his UPnP Control Point (tablet) to search for a film which is saved on the UPnP Media Server (PC) in order to watch it on the UPnP Media Renderer (STB). In order to search for the film, the user firstly needs the connection between PC and his tablet to be guaranteed by the HGW. Then, when the user has found the film saved on the PC, the STB should be turned on in order to play the film. The STB provides its display interface block, video stream decoder block, authentication block and the connection block, and the HGW assures the connection block during the entire service. This typical UPnP AV use case requires different connected devices to participate at different points in the service. When the user decides to start the service, according to the information saved about this service, the power management sends control messages to each device as they are required. The request moment information can be pre-saved by the user or the device manufacturer in the power management, or by a process of auto-learning in the power management. With the refined system, only the required

components are turned on, and the components which are no longer needed when the service is terminated are turned off.

3) Low energy communication overlay network: On each home network device, we propose an overlay low energy network by considering the characteristics of the device. The control message can be sent via ZigBee, Bluetooth Low Energy (BLE) or a UPnP low consumption network, depending on the capacity of the device. For example, it is possible that a new generation tablet will be equipped with BLE instead of having to add a ZigBee dongle to this tablet. The power consumption of a ZigBee module or BLE chipset is about a few milliwatts, which is much less than that of an Ethernet or WiFi network card which consume about 1.5 Watt. In our system, we assume that the refined power control messages will be sent by a ZigBee or BLE module in order to ensure a low-power and always-on network.

B. Refined Overlay Power Management & Refined Overlay Auto-Learning Power management

Our first proposition is the Refined Overlay Power Management (ROPM). Based on refined power control management, this proposal takes into consideration the fact that the refined power management system already has pre loaded information about the different services. The information indicates an average time lapse of each functional block for each service. The time lapse $D_{time-lapse}$ of a device functional block for a service is the time between the beginning of the service until the time when the functional block is requested. This average time lapse is a average value of user behavior for this service, and it cannot be exactly the same as user behavior. Indeed, the ROPM algorithm may turn on the functional blocks early or late. If the ROPM turns the functional blocks on early, the functional blocks will stay on until actually needed. Otherwise, if the control decision is too late compared to the actual need, the functional blocks will be turned on immediately after the ROPM detects the request by using technologies like tsocks. Tsocks is a library which transparently enables interception of outgoing messages. If the ROPM detects that the functional blocks are required, it will turn the functional blocks on immediately and ignore the decision which is too late.

Our second proposed solution is the Refined Overlay Auto-Learning power management (ROAL). It is not always possible to assume that the power management has an existing and perfect knowledge of user behaviors or device and service usages. Thus, it is difficult to predict the time when a functional block should be turned on. Therefore, we propose the Refined Overlay Auto-Learning power management (ROAL), which is able to learn when to turn the functional blocks on. When a service is launched for the first time, the ROAL turns on all the functional blocks which are necessary for the first service launch. During service execution, the ROAL gathers the information of when the functional blocks are actually requested, compares this gathered value to the saved information relating to former executions of this service, and calculates the average time lapse for each functional block: $D_{AL-time-lapse}$.

TABLE I
NOTATION SUMMARY

Notation	Definition
Nb_n^d	The number of devices in the home network
i	$i \in \{Nb_n^d\}$
Nb_d^{FB}	Number of available FB in one device
j	$j \in \{Nb_d^{FB}\}$
$FB(i,j)$	j^th functional block (FB) in i^th device.
Nb_s	Number of service repetitions.
k	$k \in \{Nb_s\}$
$service(k)$	k^{th} service instance
$P_{FB}(i,j)$	Power consumption of the Functional Blocks (FB)

III. SERVICE AND FUNCTIONAL BLOCK PATTERNS VS. ENERGY AND DELAY MODELS

In this section, we describe our service and device patterns and the models of our energy and delay calculations.

A. Service and function block patterns

A typical collaborative service pattern, which occurs repeatedly, within two devices is shown in Fig. 2 (below). Our notation is in TABLE 1. In Fig. 2 there are two instances of this service pattern $(k = 1\ to\ 2)$. Each service instance requires four Functional Blocks $FB(j = 1\ to\ 4)$ of device $i = 1$, and for device $i = 2$, the service requires three of its $FB(j = 1\ to\ 3)$. The moment $t_{request}(i,j,k)$ defines that $FB(i,j)$ has been requested in the kth service instance. The duration $D_{utilization}(i,j,k)$ defines the utilization duration of $FB(i,j)$ in the kth service instance. The duration $D_{time-lapse}(i,j,k)$ defines the interval time between the request of the $FB(i,j)$ required later and the beginning of the kth service. The inter arrival time between $k^th$and $(k+1)^th$service instances is defined as $D_{ser-inter-arrival}(k)$.

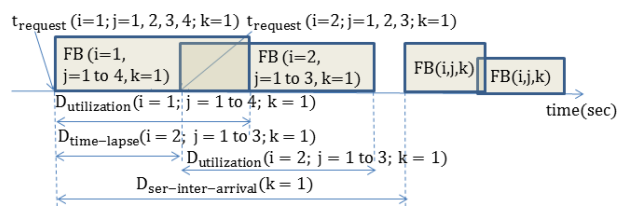


Fig. 2. A service pattern example

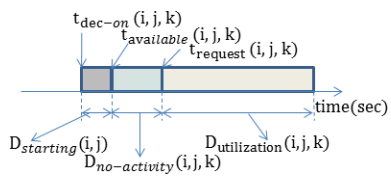
We assume that the duration of the service inter arrival follows an exponential distribution, the mean value of which is $D_{service-inter-arrival}(k)$ as described in formula (1) (below). In formula (2), the real value of each $D_{time-lapse}(i,j,k)$ follows an exponential distribution, the standard deviation of which is $\frac{1}{\lambda_0(i,j,k)}$ and the mean value of which is $\overline{D_{time-lapse}(i,j,k)}$. The standard deviation $\frac{1}{\lambda_0(i,j,k)}$ describes user behavior which may turn on the $FB(i,j)$ more or less early or late around the time of the $\frac{1}{\lambda_0(i,j,k)}$ in the kth service, with a standard deviation $\frac{1}{\lambda_0}$. The duration of the utilization of each functional block also follows an exponential distribution, the mean value of which is $\overline{D_{utilization}(i,j,k)}$ as described in formula (3).

$$D_{ser-inter-arrival}(k) = exp(\frac{1}{D_{service-inter-arrival}(k)}) \quad (1)$$

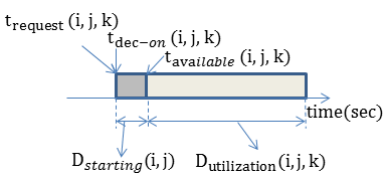
$$D_{time-lapse}(i,j,k) = t_{request}(i=1,j,k) - t_{request}(i,j,k)$$
$$= exp(\lambda_0(i,j,k)) - \frac{1}{\lambda_0(i,j,k)} + \overline{D_{time-lapse}(i,j,k)} \quad (2)$$

$$D_{utilization}(i,j,k) = exp(\frac{1}{\overline{D_{utilization}(i,j,k)}}) \quad (3)$$

Fig. 3.(a) shows the FB pattern during the service. The $t_{dec-on}(i,j,k)$ is the moment that FB(i,j) receives a decision to be turned on. The $D_{starting}(i,j)$defines the necessary starting time before $FB(i,j)$ is operational. The $t_{available}(i,j,k)$ is the moment that $FB(i,j)$ is available in the $service(k)$. If the moment $t_{available}(i,j,k)$ comes before the $t_{request}(i,j,k)$, this means that $FB(i,j)$ has a period of no activity where $D_{no-activity}(i,j,k) = t_{request}(i,j,k) - t_{available}(i,j,k)$. On the contrary, as shown in Fig. 3.(b), if the $t_{request}(i,j,k)$ comes before the moment $t_{available}(i,j,k)$, $FB(i,j)$ will start the service execution immediately upon becoming available without having a no-activity period. The duration of utilization of $FB(i,j)$ in $service(k)$ is the duration of utilization $D_{utilization}(i,j,k)$ as explained in the service pattern.



(a) Functional block is turned on in advance



(b) Functional block is turned on by request

Fig. 3. Functional block pattern

B. Energy and delay models

The energy consumption of $service(k)$ can be calculated using the following formula (4):

$$E(k) = \sum_{i=1}^{Nb_n^d} \sum_{j=1}^{Nb_d^f} P_{FB}(i,j) \times D_{on}(i,j,k) \quad (4)$$

Where the $D_{on}(i,j,k)$ is the on duration of the functional block in formula (5), which is composed of the starting duration, the utilization duration and the no-activity duration which may be 0 in the event that service request $t_{request}(i,j,k)$ is earlier than the FB being available at $t_{available}(i,j,k)$, as explained in the FB pattern.

$$D_{on}(i,j,k) = t^{pm}_{dec-on}(i,j,k) - t^{pm}_{dec-off}(i,j,k) \quad (5)$$

The $Delay(k)$ is the total waiting time for the service(k), described in formula (6). There are two cases: if the $FB(i,j)$ is available ($t_{available}(i,j,k)$) before the arrival of the service request $t_{request}(i,j,k)$, there is no waiting time for the user. Thus, the delay is nil in this case. Otherwise, if the service request arrives before the FB is available, the waiting time is the period from the $t_{request}(i,j,k)$ to the $t_{available}(i,j,k)$. The delay is the difference between these two moments.

$$Delay(k) = \sum_{i=1}^{Nb_n^d} \sum_{j=1}^{Nb_d^f} \begin{cases} 0, t_{request}(i,j,k) < t_{available}(i,j,k) \\ t_{request}(i,j,k) - t_{available}(i,j,k), else \end{cases} \quad (6)$$

After modeling our service patterns, function block patterns, and energy and delay calculations, we will apply our ROPM and ROAL propositions and other power management systems to the model.

IV. Power Management Modeling

In this section, we firstly describe our two propositions and two usual solutions as comparisons.

A. User control power management

We assume that without the help of technology, users control all their home network devices manually and individually for energy saving: they turn the device on when they need it and they turn the device off at the end of the utilization. The main inconveniences are that the user needs to wait for the functional block starting time (which includes component lighting time, booting time, etc.) and that when the device is turned on, all included $FB(1\ to\ Nb^d_{FB})$ are turned on integrally. Formula (7) defines the decision of turning on the $FB(i,j)$ in $service(k)$ made by the user control power management.

$$t^{user-control}_{dec-on}(i,j,k) = t_{request}(i,j,k) \quad (7)$$

The service is requested at the same moment that the user decides to turn the device on. Since $FB(i,j)$ is always available after the service request, the duration $D_{on}(i,j,k)$ of the $FB(i,j)$ is composed of $D_{starting}$ and $D_{utilization}$. At the end of the service, the user will turn off the device manually. The main inconvenience is that it is not automatic and could be tedious for users. Secondly, users may not think ahead and therefore have to wait for the starting time of each FB before using them. Our proposition does not have these two major drawbacks.

B. PCE power management

The PCE power management will not turn on all the functional blocks of a device integrally, but all blocks that are required are turned on at the beginning of the service. Formula (8) defines the PCE decision to turn on all necessary $FB(i,j)$ at the beginning of the service.

$$t^{pce}_{dec-on}(i,j,k) = t_{request}(i=1,j,k) \quad (8)$$

Fig. 4 (below) shows that the PCE power management turns on both device 1 and device 2 at the beginning of the service. So $FB(i=2,j)$ in device 2 stays in a no-activity state until it is actually requested. The only delay of the PCE power management is the starting time when device 1 is turned on.

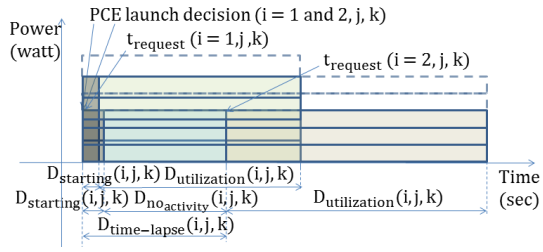


Fig. 4. PCE power management controls a collaborative service

C. ROPM & ROAL power management

Based on the same service pattern, we model our two propositions, namely ROPM and ROAL. The ROPM has knowledge of the average $D_{time-lapse}$ of the functional blocks that are required later, while the ROAL learns the time lapse $D_{AL-time-lapse}$ of the functional blocks that are required later in the service execution, described in formula (9). Formula (10) describes that at the beginning of the service, $FB(i,j)$ that are required first are turned on upon the request $t_{request}(i,j,k)$. If the refined power management does not detect a request of FB(i,j)required later, it will turn on $FB(i,j)$ required later at the moment that power management has knowledge ($\overline{D_{time-lapse}(i,j,k)}$) or $D_{AL-time-lapse}(i,j,k)$) as opposed to at the beginning of this service ($t_{request}(i = 1,j,k)$). The $D_{time-lapse}(i,j,k)$ calculates in formula (2). However, if the power management detects a request for any $FB(i,j)$ required later, it will be turned on immediately.

$$D_{AL-shift-time}(i,j,k) = \frac{\sum_{k=1}^{Nb_s} D_{time-lapse}(i,j,k)}{Nb_s} \quad (9)$$

$$t_{dec-on}^{ROPM/ROAL}(i,j,k) = \begin{cases} t_{request}(i=1)^{ROPM} + \overline{D_{shift-time(i,j,k)}} - D_{starting}, \\ \qquad t_{request}(i) \text{ is not detected} \\ t_{request}^{ROAL}(i=1) + \overline{D_{AL-shift-time(i,j,k)}} - D_{starting}, \\ \qquad t_{request}(i) \text{ is not detected} \\ t_{request}^{ROPM/ROAL}(i,j,k), \text{ Decision is late or i = 1:} \end{cases} \quad (10)$$

Fig. 5 shows that our proposition (ROPM or ROAL) decides to turn the second device on in advance. In this case, device 2 is on without executing any activity. Fig. 6 shows that our proposition (ROPM or ROAL) decides to turn on the second device too late. When the ROPM or ROAL detects that device 2 is needed immediately, device 2 is turned on at the moment that the request is detected. The delay for device 2 is the starting time of device 2. Since device 2 is turned on afterwards, it finishes its task later than the service expectation.

V. Setup of Simulations And Analysis of Results

In this section, we firstly present the simulation setup and then the analysis of the results obtained.

A. Setup of simulations

In order to accurately measure power consumption and the waiting delay of each power management, we implemented a typical home network, which was capable of executing

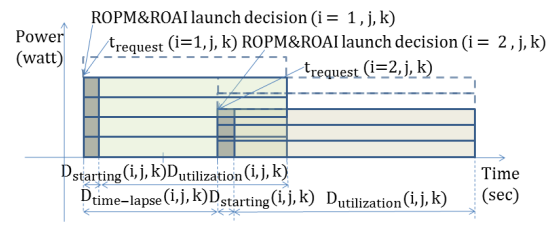


Fig. 5. Device 2 is available afterwards via ROPM & ROAL control

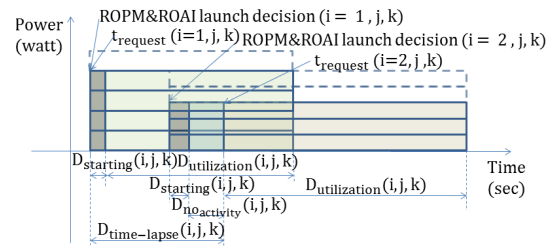


Fig. 6. Device 2 is available early via ROPM & ROAL control

a collaborative service in omnet++. The service pattern is described in Section III part A (above). The parameter values are given in TABLE II. We compare our propositions using User control and PCE power managements:

User control power management: We take a user who is mindful of energy conservation. This user turns on each device when its service is needed, and turns off each device when the service is no longer required.

PCE: The service is started with all necessary power control elements on at the beginning of the service. They will be turned off when the user finishes using the functional blocks.

ROPM: Based on the pre-saved knowledge of user habits, the functional blocks of each device in the service will be turned on immediately before the FB is needed.

ROAL: The required functional blocks are turned on when the service is requested for the first time. The ROAL learns the value of $D_{AL-time-lapse}$ of each functional block during the service execution. After obtaining this information, the functional blocks will be controlled as in the ROPM.

B. Analysis of results

In this section, we analyze the three sets of simulation results: The first study shows that the power management ROAL is able to learn an approximate accurate time laps:$D_{AL-time-lapse}(i,j,k)$, when the simulation lasts for a long time. The second study shows the energy efficiency and delay impact of each power management system, when increasing the average time lapse and standard deviation of

TABLE II
Simulation setup

Notation	Value
Nb_n^d	=2
Nb_d^f	=10
$\overline{D_{service-in-arrival}(k)}$	= 5000 s
$\overline{D_{utilisation}(i,j,k)}$	= 1000 s
$D_{starting}(i,j,k)$	= 100 s
$P_{FB}(i,j,k)$	= 10 watt
$Simulationrepetitionsforeachexperiment$	= 10 runs

user habits. The third study shows the energy efficiency and delay impact when decreasing the standard deviation of user habits with a fixed average time lapse.

1) Simulation time limit: The present study was designed to demonstrate that the ROAL can learn an accurate time lapse, when the simulation duration is long enough. The simulation time limit varies from 10 hours to 500 hours, with steps of 10 hours. We carried out 10 runs for each different simulation time limit. In Fig. 7, each point on the line red is the value of $D_{AL-time-lapse}$ that the ROAL learned at the end of each simulation. When the simulation lasts for only 10 hours, the $D_{AL-time-lapse}(i,j,k)$ is far from the $D_{time-lapse}(i,j,k)$ and the result of each run varies considerably, from 2300 seconds to 7000 seconds. However, when the simulation duration increases to about 200 hours, the ROAL obtains an accurate value of the $D_{AL-time-lapse}$ The result of the study indicates that if the ROAL has a sufficient learning period duration (200 hours, less than one week), it can learn an accurate value of time lapse of user habits. Since this value will impact the $t_{dec-on}^{ROPM/ROAL}(i,j,k)$ when the simulation limit time is increased, the power consumption and waiting delay of the ROAL power management will be impacted as shown in Fig. 8 and Fig. 9.

Fig. 8 and Fig. 9 show the average energy consumption and the average delay per service when the simulation duration is varied. In Fig. 8, when the simulation lasts for just 10 hours, the ROAL has an energy consumption value between the PCE and the ROPM. Without an accurate $D_{AL-time-lapse}$,the ROAL turned the device on in advance. Therefore, the ROAL has a higher energy consumption when the simulation lasts for a short duration. Until the simulations last for more than 200 hours, the average energy consumption is almost the same as the ROPM. In Fig. 9, when the simulation lasts for only 10 hours, the ROAL has a waiting delay value between the PCE and the ROPM. Because the ROAL does not have enough time to learn the accurate time lapse, the $D_{AL-time-lapse}$ is lower than the user habit. In cases where the functional blocks which are required later are frequently turned on in advance, there is no delay if they are already available before the service request arrives. Therefore, the delay of the ROAL approaches

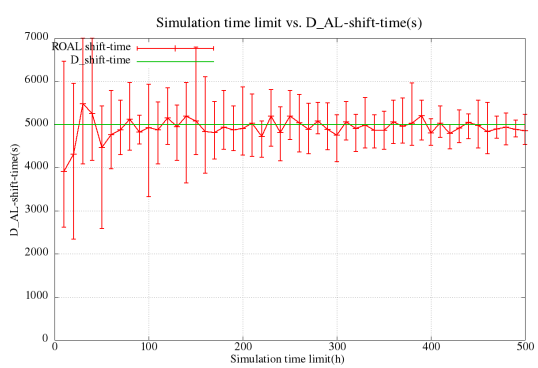


Fig. 7. ROAL progress the knowledge on the time lapse by increasing the simulation time

that of the PCE. When the simulation lasts for more than 200

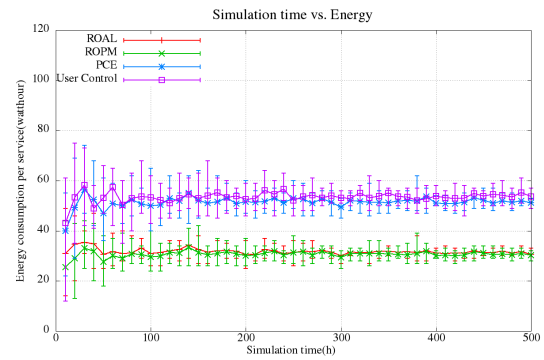


Fig. 8. Energy consumption per service by increasing simulation duration

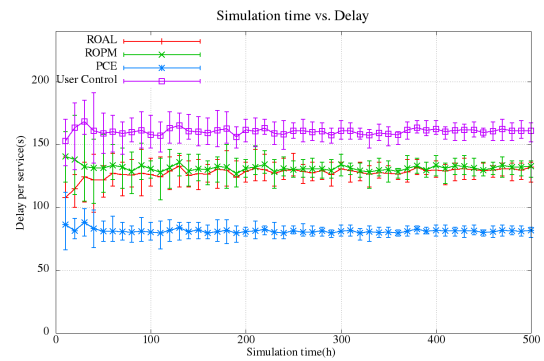


Fig. 9. Delay per service by increasing simulation duration

hours, the delay will become closer to the delay of the ROPM. From this study, we can conclude that when the simulation lasts long enough, the ROAL may obtain approximate time lapse information $D_{AL-time-lapse}$ on which the pre-saved $\overline{D_{time-lapse}}$ is in the ROPM, and the ROAL has the same energy saving performance and waiting delay as the ROPM once the information relating to the average $\overline{D_{time-lapse}}$ has been obtained.

2) Energy consumption and waiting delay by varying time lapse: In this scenario, the time lapse is varied from 0 to 5000 seconds, in steps of 100 seconds and with a standard deviation $\frac{1}{\lambda_0} = D_{time-lapse}$. As explained in Section III, the $D_{time-lapse}$ is follows an exponential process as in formula (2). When standard deviation $\frac{1}{\lambda_0} = D_{time-lapse}$, the formulae can be simplified as in formula (11). This means that the generated user habits have a mean value of $D_{time-lapse}$, but the difference between the generated time lapse and the average value will increase with the increase of the standard deviation. The simulation time limit is set at 40 hours. This is a simulation time limit for which the ROAL has not learned an accurate time lapse. Thus, we still have a difference between the ROPM and ROAL decisions.

$$D_{time-lapse}(i) = exp(\frac{1}{D_{time-lapse}(i,j,k)}) \qquad (11)$$

In Fig. 10, we can see that the energy consumption of the user control power management is almost stable because the devices are turned on integrally when the service requirements

arrive and the devices are turned off integrally when the services are terminated. Therefore, the energy consumption of the user control power management corresponds to the service utilization. The energy consumption of the PCE power management increases continuously, since the PCE turns on all participating functional blocks from the beginning of the service. The more the $D_{time-lapse}$ increases, the more energy the functional blocks consume in the no-activity period. So, with the PCE, the total energy consumption in one service increases as the $D_{time-lapse}$ increases. The ROPM consumes less energy compared to the PCE because, once one service begins, those functional blocks that are not necessary will not be launched at the beginning of the service. They are launched at the time that is pre-saved in the power management. Since the average time lapse $D_{time-lapse}$ is fixed, it is possible that one $FB(i,j)$ is turned on earlier. The ROAL consumes less energy compared to the PCE and slightly more energy compared to the ROPM. Unlike the ROPM, the ROAL does not have the knowledge when the functional blocks required later are requested by the service. The ROAL power management begins like the PCE and turns on all functional blocks. The more times the ROAL carries out the service, the more accurately it can learn the mean value of the $D_{time-lapse}$ if the average user behavior does not change. Here, during a 40-hour simulation launch time, the ROAL learns to turn on the functional blocks required later at a time that is closer to the real time lapse but which is still not accurate enough. In this scenario, the ROPM and the ROAL have a 35.24% and 41.85% energy gain respectively, compared to the PCE power management. The ROPM and the ROAL can reach 37.11% and 43.51% energy gain respectively, compared to the user control power management.

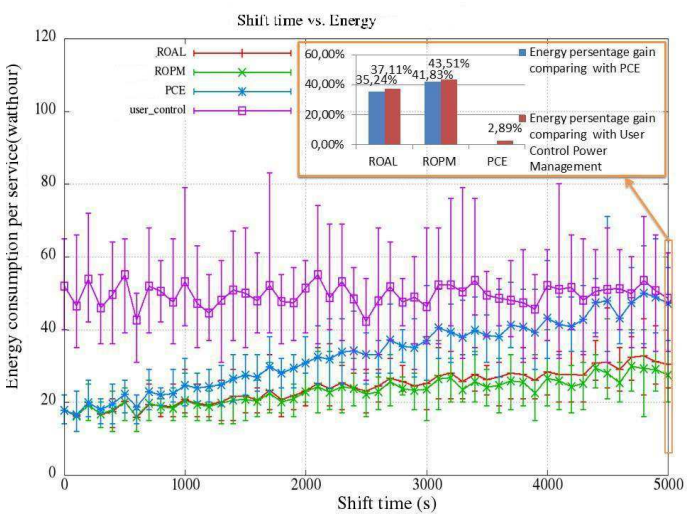


Fig. 10. Energy consumption while changing Duration of shift time

Fig. 11 shows the average waiting delay for each service. The user control power management turns on the devices when the service request arrives. Thus, there is always a waiting delay in the starting time before the device becomes available. The PCE has the smallest waiting delay because all of the

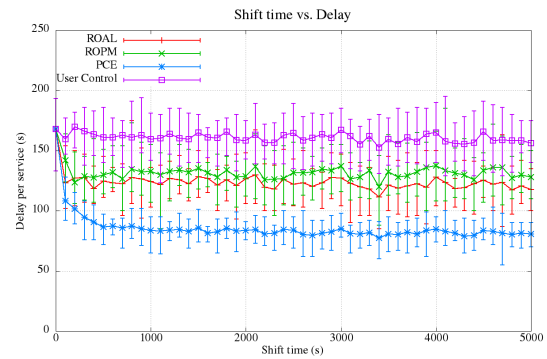


Fig. 11. Energy consumption while changing Duration of shift time

functional blocks are turned on at the beginning of the service. The waiting delay of the PCE comes from the functional blocks that are used first. Although some of the functional blocks are needed later, they are already turned on.

Therefore, the PCE has the smallest delay and it is impossible to have a smaller delay, unless the functional blocks are never turned off. The PCE could be seen has an ideal form of management from the point of view of the delay, but it is the least advantageous from the power saving point of view (cf. Fig. 10). The ROPM has a greater waiting delay than the PCE. At the beginning of the service, there is always a waiting delay for the functional blocks that are used first. However, the functional blocks that are used later are turned on according to the predicted usual user behavior. If the functional blocks are turned on early, there is no generated delay. In contrast, it is also possible that the decision to turn on is later than the functional block requirements. When the ROPM detects the requirements of functional blocks, it turns the functional blocks on immediately. In this late decision case, the ROPM generates an extra delay compared to the PCE. The ROAL has a smaller waiting delay than the ROPM because the ROAL has an inaccurate learned time lapse which is earlier than user habit. There is a greater likelihood that the ROAL will turn $FB(i,j)$ on earlier than the average service request. If $FB(i,j)$ is available before the request, there is no waiting delay for the user. The result of this study indicates that the ROPM is the most energy efficient system and has a smaller delay than the user control power management, but a greater delay than the ROAL and the PCE. The ROAL is second in terms of energy efficiency but has a smaller delay compared to the ROPM.

3) Energy consumption and waiting delay by varying standard deviation : In this section we set the $D_{time-lapse}$ at 5000 seconds and decrease the standard deviation $\frac{1}{\lambda_0(i,j,k)}$ of the requirements of functional blocks which follows an exponential distribution as in formula (12). In this section we fix the $\overline{D_{shift-time}}$ at 5000 seconds and vary the standard deviation λ_0 of the requirements of functional blocks which follows an exponential distribution.

$$D_{time-lapse}(i) = exp(\lambda_0(i,j,k)i) - \frac{1}{\lambda_0(i,j,k)} + 5000 \quad (12)$$

When the standard deviation $\frac{1}{\lambda_0(i,j,k)}$ is decreased, the user habits for turning on the functional blocks needed later gets increasingly closer to $D_{time-lapse} = 5000$. The standard deviation of the time lapse decreases from 500 to 0 seconds with steps of 10 seconds. In Fig. 12, the energy consumption of the four power management systems are stable. The power consumption of the user control power management depends on the utilization of the device where the average value is fixed. Thus, the power consumption of the user control power management will remain stable. The PCE power consumption stays stable because the power-on duration of $FB(i,j)$ which corresponds to utilization and time lapse is stable. The decision of the ROPM power management takes into account the value of $D_{time-lapse}$ and the ROAL takes into account the value of $D_{AL-time-lapse}$. These two values will not change when the standard deviation is varied. Consequently, functional blocks are turned on based on fixed values in the ROPM and ROAL and the energy consumption stays stable as the last point $D_{time-lapse} = 5000$ in Fig. 10. Regarding the delay shown in Fig. 13, we can see that the delay of the user control and PCE power management systems stay stable. The ROPM and ROAL delay decreases when the standard deviation is decreased. Since the standard deviation decreases, this means that the user behavior is approaching the pre-loaded time lapse or learned time lapse. The prediction of our power managements decision could be more accurate. Thus, the power management systems have a smaller delay impact. We can draw from this study that standard deviation is an important user habit which could have an impact on the delay.

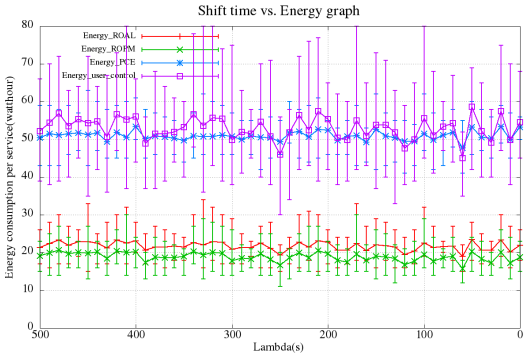


Fig. 12. Energy consumption when standard deviation is changed

VI. Conclusion

In this paper, we propose two power management systems based on refined overlay power management: the ROPM and ROAL. Our two propositions are based on the services analysis by controlling the functional blocks. The ROPM is energy efficient with the help of the pre-saved user behavior information. The ROAL is able to obtain approximate user behavior information following a learning period and it has a facility for the implementation in the home network. The simulation results has shown that ROAL provides a same level of energy efficiency comparing with ROPM. The ROPM and ROAL can reach 37.11% and 43.51% energy gain respectively.

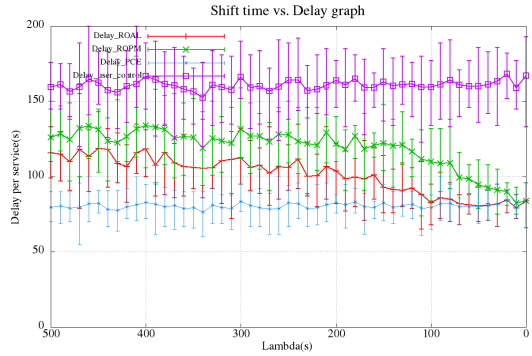


Fig. 13. Delay when standard deviation is changed

Therefore, after analyzing the impact of the standard deviation on the delay, our future research will explore the auto-learning of the standard deviation of the user behavior probability distribution in order to retrieve the trade-off between waiting delay and energy efficiency.

References

[1] B. Rose, "Home networks: a standards perspective," *IEEE Communications Magazine*, vol. pp. 7885, 2011.

[2] S. E. Eurostat, "Consumption of energy," Tech. Rep., 2012.

[3] "European residential energy pricing report 2013," Vaasa ETT, Tech. Rep., 2013.

[4] L. Benini, R. Bogliolo, and G. D. Micheli, "A survey of design techniques for system-level dynamic power management," *IEEE Transactions on VLSI Systems*, vol. 8, pp. 299–316, 2000.

[5] E.-Y. Chung, L. Benini, A. Bogiolo, and G. De Micheli, "Dynamic power management for non-stationary service requests," in *Proceedings of the conference on Design, automation and test in Europe*, ser. DATE '99. New York, NY, USA: ACM, 1999.

[6] Maruti, "Dynamic ethernet link shutdown for energy conservation on ethernet links," in *ICC*, 2007, pp. 6156–6161.

[7] B. Khargharia, S. Hariri, and M. S. Yousif, "An adaptive interleaving technique for memory performance-per-watt management," *IEEE Trans. Parallel Distrib. Syst.*, vol. 20, no. 7, pp. 1011–1022, Jul. 2009.

[8] X. Fan, C. Ellis, and A. Lebeck, "Memory controller policies for dram power management," in *Proceedings of the 2001 international symposium on Low power electronics and design*, ser. ISLPED '01. New York, NY, USA: ACM, 2001, pp. 129–134.

[9] Y.-K. Jeong, I. Han, and K.-R. Park, "A network level power management for home network devices," *Consumer Electronics, IEEE Transactions on*, vol. 54, no. 2, pp. 487–493, May 2008.

[10] J. Ritchie and T. Kuehnel, *UPnP AV Architecture:1*, The UPnP forum Std., 2002.

[11] H. Yan, C. Gueguen, B. Cousin, J. P. Vuichard, and G. Mardon, *"Green Home Network based on an Overlay Energy Control Network"*, J. L. M. Shafiullah Khan, Ed. CRC press, Taylor & Francis Group, USA, 2013.

[12] H. Yan, F. Fontaine, O. Bouchet, J.-P. Vuichard, J.-P. Javaudin, M. Lebouc, M.-H. Hamon, B. Cousin, and C. Gueguen, "Hope: Home power efficiency system for a green network," Turin, Italy, 2013.

[13] S. Farahani, *ZigBee Wireless Networks and Transceivers*. Newton, MA, USA: Newnes, 2008.

[14] *UPnP Low Power Architecture*, The UPnP forum Std.

[15] J. Hui and D. Culler, "Extending ip to low-power, wireless personal area networks," *Internet Computing, IEEE*, vol. 12, no. 4, pp. 37–45, July 2008.

[16] E. Mackensen, M. Lai, and T. Wendt, "Bluetooth low energy (ble) based wireless sensors," in *Sensors, IEEE*, Oct 2012, pp. 1–4.

[17] M. Flannery, "Method and apparatus of providing power management using a self-powered universal serial bus (usb) device," Aug. 25 1998, 5,799,196 US Patent.

[18] C. Ryu, "Method and apparatus for controlling power of computer system using wake up lan (local area network) signal," Jul. 8 2003, 6,591,368 US Patent.