Machine-to-Machine Communication over TV White Spaces for Smart Metering Applications

Luca Bedogni, Angelo Trotta, Marco Di Felice, Luciano Bononi

September 10, 2013

Abstract

Machine-to-Machine communications is envisioned to become one of the fundamental pillars of the future Internet of Things paradigm, enabling platoons of devices to be seamlessly connected and to cooperate over smart spaces. Among the possible application scenarios, smart metering represents an already existing technology that might take benefit from the capability of autonomous configuration and setup of M2M networks. At present, smart meters communicate over the 2G/3G network, however the utilization of the cellular technology poses several problems, such as low coverage and spectrum shortage over dense areas. To overcome these issues, in this paper we investigate the application of cognitive radio principles over TV White Spaces to M2M communication for the smart metering scenario. Following the recent regulations of FCC and Ofcom, that foresees the presence of a spectrum database for TV white spaces detection, we study the trade-off between protection of licensees and energy consumption in a cluster of smart meters. We provide three novel research contributions: (i) an analytical model to estimate the lifetime of a cluster of smart meters; (ii) centralized and distributed algorithms to determine the schedule operations of Master/Slave devices foreseen by the spectrum regulations; (iii) performance evaluation of the proposed framework through extensive Omnet++ simulations.

1 Introduction

It is a common thought that M2M communication will assume a key role for wireless networks deployment as soon as the Internet of Things [2] and the Smart Grid [7] paradigms will be progressively deployed in everyday life. Different from traditional Human-to-Human (H2H) and Human-to-Machine (H2M) communication systems, M2M have some unique characteristics, such as short packets size, bursty traffic and self-configuration capabilities. Among the possible applications, smart metering constitutes a prominent research field, that can leverage these characteristics. Nowadays, smart meters are already deployed by using the
2G/3G cellular technology as communication infrastructure. However, this solution presents some drawbacks, like the low coverage in certain areas, given the fact that smart meters are typically placed underground, with heavy shadowing effects caused by walls and other buildings. In addition, cellular bands already experience a heavy utilization over dense areas, so they might not be suitable to support a large scale deployment of M2M networks.

In this paper we investigate the benefits of using cognitive radio technologies [1] over TV white spaces for M2M communication. Recently, with the digital TV switchover, several national spectrum regulators are considering opportunistic spectrum access techniques by secondary users in the VHF and UHF bands, commonly referred to as TV White Spaces [9] [13]. These frequencies range from 30 up to 862 MHz, with some differences from country to country, and have unique propagation and penetration through obstacles characteristics, making them an ideal candidate to solve the challenges presented by the smart metering scenario.

In this paper, we mainly refer to regulations made by FCC in the U.S. [9] and Ofcom in the U.K. [13], in which it is foreseen that cognitive devices willing to access the TV spectrum should firstly query a remote database in order to get the list of available channels at their positions. These devices, equipped with geo-localization capabilities, are called Mode II [9] or Master [13]. Moreover, regulators foresee the possibility of another class of devices called Mode I [9] or Slave [13] which do not have direct access to the remote database, but can acquire spectrum information from Master/Mode II devices. Since smart meters are naturally clustered, for example in blocks of apartments and in residential buildings, they can leverage the spatial proximity in order to reduce the database queries and prolong battery life. In this work we investigate the trade-off between database query load and smart meters battery life, by analyzing the Master/Slave scheduling operations in each cluster. We propose three novel contributions in this area. First, we present an analytical model to derive the cluster lifetime, modeling the energy consumption of the transmissions occurring in each cluster for TV white space detection and for smart meters measurements. Through this model, we express the notion of goal satisfaction defined as the minimum battery lifetime requested to the cluster. Second, we formulate the Master/Slave scheduling as an optimization problem and we introduce centralized and distributed solutions to determine a feasible assignment of Master/Slave roles to smart meters that guarantees the goal satisfaction while minimizing the number of switchings. Finally, we evaluate our framework through extensive Omnet++ simulations that show the benefits of the proposed architecture in terms of energy consumption and communication overhead.

The rest of the paper is organized as follows: in Section 2, we describe the smart metering scenario and discuss the state of the art; in Section 3 we present the analytical model; in Section 4 we introduce the Master/Slave scheduling problem and we propose a centralized solution; in Section 5 we describe distributed scheduling protocols by taking into account different system requirements (cluster lifetime, fairness, etc.); in Section 6 we evaluate the proposals through extensive Omnet++ simulations; Conclusions follows in Section 7.
In this Section we describe the scenario of smart metering communication and we review the state of the art. In Figure 1 a typical smart metering scenario is depicted. Here, multiple smart meters need to report their sensor readings to a central aggregator once per day, that is in charge of merging data and transmitting them to the utility servers. At present, this task is typically done by each smart meter through 2G/3G cellular connection. The smart meters could be main powered or battery powered, depending on the type of measurement. For instance, electrical smart meters could be main powered, so energy efficiency is not an issue, while gas smart meters are typically battery powered, thus energy efficiency plays a crucial role in terms of costs and maintenance efforts, as also stated in [15]. Most of the existing deployments of smart meters rely on the 2G/3G cellular infrastructure to communicate readings to the central aggregator, for cost and data rate reasons (due to the fact that short messages are exchanged). However, this might not be the trend for the future due to emerging spectrum demanding services for the smart grid [7] [17] [18] and the consequent spectrum shortage over the 2G/3G bands [14]. To this purpose, research is investigating solutions to switch from cellular M2M communications to capillary M2M communications [3], in which devices build a virtual mesh that is connected to a central aggregator. This architecture could be supported by the usage of lower frequencies in the UHF band, the so-called TV White Spaces (TVWS), that are currently underutilized [12] [16] and whose opportunistic access is regulated from country to country by national agencies, like the FCC in the U.S. [9] and Ofcom in the U.K. [13]. According to these rules, devices need to query a remote database to get the list of available channels in the area. Several field trials have already shown the availability of TVWS, both in rural areas and in urban areas [5] [6], and the benefits of M2M communication for smart grid applications [12] [19]. Some works have also examined the benefits
of using TVWS for smart grid enhancement, highlighting some fundamental characteristics of these new devices, such as self-organization, self-management and self-healing [19]. In [16] the authors showed the main differences in TVWS availability between the US and Europe. A test bed was set up in [4] in Scotland, demonstrating the advantages of TVWS for the smart grid scenario. In [11] the authors propose an architecture for the TVWS spectrum database. In [10] the authors reviewed the current state of the art of wireless smart metering technologies. They derived mathematical models to investigate the efficiency of different wireless technologies. However, they did not take into account TVWS as a possible candidate. In [8] the authors analyzed the benefits of using TVWS for the smart metering scenario, highlighting the bandwidth gains as well as investigating the security issues. They also introduced machine learning techniques to identify and prevent possible attacks. However, their work does not focus on battery consumption, and does not take into account the benefits of opportunistically switching between Master and Slave modes. Compared to the existing solutions, we provide the following novelties: (i) we take into account the FCC and Ofcom regulations for the Master/Slave operations; (ii) we investigate the lifetime of a cluster of smart meters, also considering the energy overhead needed for TVWS detection; (iii) we address the Master/Slave scheduling problem by proposing centralized and distributed solutions.

3 Analytical model

In this Section we present our analytical model to estimate the lifetime of the cluster of smart meters considering both the consumption due to data communications and to the overhead for TVWS detection. In addition, we introduce the notion of goal satisfaction in Section 3.1. Through our model, we then investigate the optimal scheduling of Master/Slave operations in Section 4 and 5. We assume smart meters are grouped into clusters. Each cluster is composed by a single Master Device (MD) and a number of Slave Devices (SD). Each smart meter is capable of working both as MD or as SD, depending on the scheduling policy. The MD is responsible for querying the database to get the list of TVWS in the area, and to instruct SDs about available channels. More specifically the set of actions the MD must perform every day are depicted in Figure 2. Here, six different time slots are identified: the Read slot, in which each smart meter performs the reading; the DB slot, in which the MD queries the spectrum database and communicates to SDs the available TVWS and the selected channel to be used in the cluster; the RX slot during which the MD receives the SDs readings on the selected channel; the Merge slot in which the MD puts together the readings in a single packet; the TX slot in which the MD transmits this packet to the central aggregator; finally the Manage slot in which the MD decides whether to start an election process to decide the next MD of the cluster, according to the scheduling policy. In case, the election process works as follows: the MD sends a message to all the SDs. Each SD replies
Figure 2: The actions a MD perform every day. SDs perform only the read operation and send the value to the MD at minimum power.

with its availability (given by the specific scheduling algorithm in use) on being the MD for the next day. At this point, the MD examines all the replies and decides which device will be the next MD, sending a direct message to it. For each election, a total of 2 messages are sent by the MD, while each SD sends 1 message only. For ease of modeling, we assume in this study that elections will be performed at each Manage slot, which constitutes a lower bound in terms of energy efficiency of the cluster. We will relax this assumption in Section 5. SDs are responsible only to send their smart readings to the MD, thus limiting their transmission power as stated before. Both the MD and SDs enter in a power saving mode to save battery life while not performing any of the activities stated above. In the following, we derive the energy consumed each day by a MD as the sum of the actions depicted in Figure 2. This is given by Equation 1:

\[ E_{MD} = E_{re} + E_{DB,MD} + E_{rx} + E_{ag} + E_{tx} + E_{ma,MD} + E_{id,MD} \]  

where \( E_{re} \) is the energy consumed to read the house consumption, \( E_{DB,MD} \) is the energy needed to query the remote database in order to receive the list of available channels, \( E_{rx} \) is the energy consumed with the radio in receiving mode, \( E_{ag} \) is the energy needed to merge together neighbor’s packet into the final message sent, \( E_{tx} \) is the energy consumed for sending the aggregated packet to the aggregator, and \( E_{ma} \) is the energy a MD must spend in order to manage the switching process, determining if it still need to be the MD or an election must occur. Finally \( E_{id, ch} \) is the time consumed while in power saving mode. By referring to Table 1, we derive all the single terms of Equation 1:

\[ E_{re} = r \cdot t_r \cdot c_r \]  

\[ E_{DB,MD} = 2 \cdot (t_{q,db} \cdot \gamma) + (t_{DB} - 2 \cdot t_{q,db} - (n - 1) \cdot s t_{a,SD}) \]  

\[ E_{rx} = t_{rx} \cdot \beta \]  

\[ E_{ag} = (n - 1) \cdot t_{ag} \cdot c_a \]
Table 1: The symbols used in the analytical model

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_r$</td>
<td>Time needed to perform a single sensor read</td>
</tr>
<tr>
<td>$r$</td>
<td>Number of sensor readings per day</td>
</tr>
<tr>
<td>$c_r$</td>
<td>Computational cost to pay for a single sensor read</td>
</tr>
<tr>
<td>$t_{q, db}$</td>
<td>Time needed to transmit a query to the database</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>Energy needed to transmit a packet at full power</td>
</tr>
<tr>
<td>$\delta$</td>
<td>Energy needed to transmit a packet at reduced power</td>
</tr>
<tr>
<td>$t_{db}$</td>
<td>Duration of the DB time slot</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Energy consumed while in sleeping mode</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Energy consumed while in Rx mode</td>
</tr>
<tr>
<td>$n$</td>
<td>Number of nodes</td>
</tr>
<tr>
<td>$t_s$</td>
<td>Time needed to transmit a message to the central aggregator</td>
</tr>
<tr>
<td>$t_{ag}$</td>
<td>Time needed to merge a message from a Slave device into the final packet to be sent to the aggregator</td>
</tr>
<tr>
<td>$t_{a, SD}$</td>
<td>Time needed to send a packet to the Master device to get the list of available channels</td>
</tr>
<tr>
<td>$c_a$</td>
<td>Energy consumed by the active CPU</td>
</tr>
<tr>
<td>$t_{el}$</td>
<td>Duration of the election packet</td>
</tr>
<tr>
<td>$t_{SD}$</td>
<td>Duration of the packet containing the readings to the MD</td>
</tr>
<tr>
<td>$t_{DB}$</td>
<td>Duration of the DB slot</td>
</tr>
<tr>
<td>$t_{MA}$</td>
<td>Duration of the Manage slot</td>
</tr>
<tr>
<td>$t_{idle, MD}$</td>
<td>Time a Master device passes in idle mode</td>
</tr>
<tr>
<td>$t_{idle, SD}$</td>
<td>Time a Slave device passes in idle mode</td>
</tr>
<tr>
<td>$E_{id, MD}$</td>
<td>Energy consumed in power saving mode by a Master device</td>
</tr>
<tr>
<td>$E_{id, SD}$</td>
<td>Energy consumed in power saving mode by a Slave device</td>
</tr>
<tr>
<td>$s_{day}$</td>
<td>Seconds in a day, i.e. 86.400</td>
</tr>
<tr>
<td>$E_{start}$</td>
<td>Initial amount of energy</td>
</tr>
<tr>
<td>$t_{rx}$</td>
<td>Time during which the device is in receiving state</td>
</tr>
</tbody>
</table>
\[ E_{tx} = \gamma \cdot t_s \]  

\[ E_{ma,MD} = 2 \cdot t_{el} \cdot \gamma + n \cdot c_a + (t_{MA} - 2 \cdot t_{el}) \cdot \beta \]  

\[ t_{idle,MD} = s_{day} - (n - 1) \cdot t_{ag} - r \cdot t_r - t_{DB} - t_{RX} - t_s - t_{MA} \]  

\[ E_{id,MD} = \alpha \cdot t_{idle,MD} \]  

The energy consumed by any SD is defined instead as:  

\[ E_{SD} = E_{re} + E_{DB,SD} + E_{tx,SD} + E_{ma,SD} + E_{id,SD} \]  

where \( E_{re} \) is the same as before, \( E_{DB,SD} \) is the energy for asking the MD the list of available channels, \( E_{tx,SD} \) is the energy needed to send a message to the neighbor by using the minimal transmission power, \( E_{ma,SD} \) is the energy consumed during the Manage slot by any SD, and \( E_{id,SD} \) is the energy needed to spend the rest of the time in power saving mode.  

As before, we derive all the terms in Equation 10:  

\[ E_{db,SD} = 2 \cdot t_{a,SD} \cdot \delta \]  

\[ E_{tx,SD} = \delta \cdot t_{SD} \]  

\[ E_{ma,SD} = t_{el} \cdot \delta + (t_{MA} - t_{el}) \cdot \beta \]  

\[ t_{idle,SD} = s_{day} - t_{SD} - r \cdot t_r - 2 \cdot t_{a,SD} - t_{MA} \]  

\[ E_{id,SD} = \alpha \cdot t_{idle,SD} \]  

Let \( j \) be SD or MD, and \( E_j[d] \) be the amount of energy consumed by any device of the cluster while being in mode \( j \) until day \( d \). This can be written as:  

\[ E_j[d] = E_j \cdot d \]  

Analogously we derive the energy remaining at day \( d \) for device \( i \) being in mode \( j \) as:  

\[ E_{i,j}^{left}[d] = E_i^{start} - E_j[d] \]  

where \( E_i^{start} \) is the initial amount of energy of device \( i \). As depicted in Figure 3 for a configuration with \( E_i^{start} \) equal to 16000, \( E_{i,j}^{left}[d] \) and \( E_{i,j}^{left}[d] \) describe two lines, and all possible energy consumptions fall in between the two lines. Let \( \theta_j \) be the time in which the device reaches out of energy with the current energy \( E \) while always being in mode \( j \):  

\[ \theta_j[E] = \frac{E}{E_j} \]  

It is easy to see that the lifetime of device \( i \) will fall between \( \theta_{SD}[E_i^{start}] \) and \( \theta_{MD}[E_i^{start}] \), because a device cannot consume more energy than always being a MD, and inversely, a device could not consume less energy than always being a SD.
3.1 Satisfying a goal

Based on the analytical model described so far, we introduce the notion of cluster goal, expressed in terms of battery life of the cluster. The cluster lifetime is defined as the time passed from the initial deployment till the first death of any smart meter. This is a common assumption found in literature, and is also practical from our point of view: when a battery expires, human intervention is required, and the costs to change a battery to a device or to a couple of devices near to each other are almost the same. More specifically, we define the goal as follow:

**Definition 1 (goal)** a goal is a minimal duration $\psi$ expressed in days every device in the cluster should guarantee.

We also define the notion of no return point as follows:

**Definition 2 (nrp)** given a day $d$, a no return point (nrp) $\chi_d$ is the lowest energy value for which, $\theta_{SD}[\chi_d] + d \geq \psi$.

The $\chi_d$ of any device at day $d$ simply indicates whether the device could become a MD, or if it needs to remain a SD in order to guarantee the goal. In Figure 3 we depicted a configuration of devices with the notion of goal and nrp. Since devices may have different residual energy, and the nrp could be passed with a single day as a MD, we now define the notion of soft no return point.

**Definition 3 (soft nrp)** Suppose $E_i^d[d]$ is the energy left for device $i$ at day $d$. A soft nrp is the greatest point $z_i$ for which the disequation $E_i^d[z_i] - E_{MD} \geq \chi_{z_i+1}$ holds.

Intuitively, the soft nrp $z_i$ indicates the total number of days device $i$ could serve as MD, considering that the switching operations are performed once per
day. We now define $\rho$ as the sum of the soft nrp points of the devices (i.e. $\rho = \sum_{i=0}^{\infty} z_i$). This indicates the total number of days the cluster of smart meters could be served by a MD. We can now derive the notion of satisfiability as follows.

**Definition 4 (satisfiability)** Assuming all smart meters can reach the goal if they are always in SD mode, a system is satisfiable when the cluster can reach the goal. This happens iff $\rho \geq \psi$.

Definition 4 claims that the system can reach the goal if there is a device that can serve as MD in every day till the goal $\psi$. If at a certain time all the devices reach their own soft nrp, then no MD can be elected, and therefore the system is not satisfiable.

## 4 Centralized framework

In this Section we use the analytical model presented in Section 3 to derive an optimization framework that determines the Master/Slave scheduling assignments among the devices. We formulate the scheduling as an optimization problem, in which the inputs are the number of nodes in the cluster $N$, the amount of time each device $i$ can be Master $MD_i$, and the global goal $\psi$. The output of the centralized framework is a vector $D$ where $D_i$ is the total amount of days node $i$ has to be the Master device of the cluster. More formally:

**Given:** $n, MD_i, \psi$

**To find:** $D \rightarrow \{D_0, D_1, \ldots, D_{n-1}\}$

**Subject to:**

$D_i \leq MD_i$  

$\sum D_i \geq \psi$  

$\text{minimize } \left( \max_{i \in [0,n]} \left( \frac{D_i}{MD_i} \right) - \min_{i \in [0,n]} \left( \frac{D_i}{MD_i} \right) \right)$

Here, Constraint 21 ensures that any device will not be requested to schedule more days as Master than what it can afford (given by $MD_i$). Constraint 22 guarantees that the vector $D$ satisfies the goal, according to Definition 4. Finally, Constraint 23 is the fairness condition, since we attempt to balance the effort of coordinating the cluster among all the devices. The $MD_i$ is exactly the soft nrp point of device $i$, and can be computed through geometrical reasoning as:

$MD_i = \left\lfloor \frac{E_{\text{start}} - E_{\text{SD}} \cdot \psi}{E_{MD} - E_{SD}} \right\rfloor$

Given this problem, we can show that the optimum can be reached when each smart meter $i$ contributes as MD to the goal in a proportional way, based on its own $MD_i$ value. More formally, the term $D_i$ for smart meter $i$ is computed as
follows:

\[ D_i = \left\lceil \psi \cdot \frac{MD_i}{\sum_{j=0}^{n-1} MD_j} \right\rceil \]  

(25)

We show now that this assignment is fair given Condition 23 and determines a feasible assignment guaranteeing the goal \( \psi \) if the system is satisfiable according to Definition 4. The fairness of the system is bounded in the 

\([0, \frac{1}{\min(D_i)}]\)

interval. This is straightforward following algebraic transformations in Equation 23 by substituting the \( D_i \) term with the one given in Equation 25. We prove now the satisfiability of the goal with the Master/Slave scheduling using Equation 25.

**Proof 1 (Satisfiability)** According to Definition 4, a system is satisfiable if

\[ \sum_{j=0}^{n-1} MD_j \geq \psi, \]  

then

\[ \frac{\psi}{\sum_{j=0}^{n-1} MD_j} \leq 1, \]  

so \( D_i \leq MD_i \), therefore Constraint 21 is satisfied. At the same time, it is easy to see that Constraint 22 is satisfied since

\[ \sum_{i=0}^{n-1} D_i \geq \psi. \]

## 5 Distributed Scheduling protocols

Solving the optimization problem presented in Section 4 requires central coordination, either by the aggregator or by another entity and might involve considerable computation effort. For this reason, we propose here and evaluate different distributed scheduling algorithms to approximate the optimum given by Equation 20. The first three algorithms follows straightforward approaches, but are tested here to provide lower and upper bounds on the metrics evaluated in Section 6. The last algorithm (i.e. the Cost Aware) is proposed to prolong cluster lifetime while addressing the fairness issues.

### 5.1 No election protocol

This algorithm does not take into account the optimization framework. Basically, every device acts as there is no cluster, and thus makes its own queries to the spectrum database, serving as a Master device. In this case, the final goal is satisfied iff all devices have \( E_{MD}[z] \geq \psi \). This protocol minimizes the overhead for the cluster management since no elections are performed.

### 5.2 Highest first protocol

In this algorithm, the device with the highest residual energy serves as the Master. As a result, an election is issued at the end of every day. This algorithm provides an upper bound on the overhead for the cluster management.
5.3 Greedy protocol

In this algorithm, each smart meter $i$ acts as Master for $MD_i$ days, i.e. the maximum number of days it can serve as Master device before passing the soft nrp threshold. It is easy to see that this method reduces the number of elections but does not provide the fairness of the system, since some smart meters might serve for $MD_i$ days while others may not become a Master device at all.

5.4 Cost Aware Protocol

This algorithm provides a distributed implementation of the centralized schedule given by Equation 25, while relaxing the worst case assumption taken in the analytical model. To this aim, each device $i$ maintains two values $E_{i,SD}$ and $E_{i,MD}$ representing the average energy consumption for being in SD or MD mode. We highlight that $E_{i,SD}$ and $E_{i,MD}$ may not be constant due to the fact that elections might be issued or not at each Manage slot, thus impacting the energy consumption per day of each device. At network setup, $E_{i,SD} = E_{SD}$ and $E_{i,MD} = E_{MD}$, i.e. the worst case scenario configuration is utilized, and an election is issued at the first day. At each election, smart meter $i$ updates its $MD_i$ value with Equation 24 (using $E_{SD} = E_{i,SD}$ and $E_{MD} = E_{i,MD}$), thus accounting for its actual battery consumptions and network capabilities, and sends it to the MD. Based on these values, the current MD populates the vector $D$ by using Equation 25 and chooses the smart meter $j$ with maximum value of $D_j$. At this point, the smart meter $j$ will be in MD mode for exactly $D_j$ days, till issuing another election with the same modalities described above.

6 Performance evaluation

In this Section, we evaluate the distributed protocols proposed so far, by considering the following metrics for the comparison:

- **Cluster lifetime**: this is defined as the time elapsed from the network setup till the instant of the first death, i.e. when the first smart meter runs out of battery.

- **Goal satisfaction**: fixing a goal and a number of simulation run for a specific network configuration, this is the percentage of runs in which the goal is satisfied by the cluster.

- **Fairness**: this is a measure of the workload balance among the devices and it is defined through Equation 26.

- **Elections**: this is defined as the total number of elections issued in the network till the goal is reached. Practically, it measures the overhead involved by each scheduling algorithm for cluster management operations.
For the evaluation, we model a smart metering scenario within the Omnet++ tool, and we vary the number of devices in the network. Each smart meter communicates at full transmitting power $\gamma$ (to reach the central aggregator) and $\delta$ (to reach its neighbors). Nodes are heterogeneous and are provided with different initial energies ($E^{\text{start}}_i$) and energy consumption values (i.e. $E_{SD}$ and $E_{MD}$) while being a Slave or Master device. In each simulation run, these values are randomly generated in the interval given by Table 6.

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial energy</td>
<td>$8.000 \text{ Ah} \rightarrow 12.000 \text{ Ah}$</td>
</tr>
<tr>
<td>$n$</td>
<td>$2 \rightarrow 200$</td>
</tr>
<tr>
<td>$E_{SD}$</td>
<td>1</td>
</tr>
<tr>
<td>$E_{MD}$</td>
<td>$(1 \rightarrow 3) + N \cdot c_a$</td>
</tr>
</tbody>
</table>

6.1 Cluster lifetime

Figure 4: In Figure 4 we show the cluster lifetime using different algorithms.

Figure 4 shows the total cluster lifetime using the different scheduling algorithms as a function of the number of smart meters in the network. It is immediately evident that using no elections at all gives the worst results in terms of cluster lifetime. Inversely, the battery lifetime is greatly improved by using the Cost aware and Greedy algorithms, which shows a similar trend in terms of trade-off between the number of devices and the battery lifetime. In fact, for a number of devices lower than 10, the lifetime of the cluster increases with the number of devices since the MD overhead is shared among more devices. At the same time, this benefit is reduced as the number of devices is greater than 10, since the $E_{MD}$ cost is proportional to the cardinality of the cluster. Although the aggregated performance of the Cost aware and Greedy algorithms are similar,
we will show the benefits provided by the first solution in terms of fairness in Section 6.3.

6.2 Goal satisfaction

In this Section we keep constant the number of devices and we evaluate the ability of the proposed algorithms to satisfy a pre defined goal. Figure 5(a) and Figure 5(b) depicts the percentage of simulations runs in which the goal (on the x-axis) is satisfied, for a configuration with 10 and 500 devices respectively. We considered 1000 runs for the evaluation with the parameters of Table 6. As for the previous analysis, the Cost aware and the Greedy algorithms provide better performance, achieving a higher percentage of satisfied goal compared to the other two algorithms. Both figures show the same trend, although with different slopes due to the impact of the number of devices on the power consumption of the cluster.

6.3 Fairness

In this Section we evaluate the work balance between the smart meters of the scenario by considering the fairness index $F$ computed as follows:

$$F = 1 - \left( \max_{i \in [0,n]} \left( \frac{D_i}{MD_i} \right) - \min_{i \in [0,n]} \left( \frac{D_i}{MD_i} \right) \right)$$

(26)

Here, $D_i$ is the amount of time a device has been the master, and $MD_i$ expresses its MD capacity according to Equation 24. An $F$ value close to 1 indicates a higher fairness in balancing the Master/Slave scheduling operations, while a value close to 0 has to be intended as a greater diversity between work loads of the devices, and thus a low fairness. In Figure 6(a) we show the $F$ metric as a function of the number of devices, for three scheduling algorithms (we omit the No election protocol since each device acts as MD, and thus no switching operations occurs). Here, the Greedy algorithm experiences the worst performance, due to the fact that in some configurations devices with low $MD$ values are excluded from the cluster management. Similarly, the same behavior is shown with the Highest first algorithm, where however devices with higher battery life contribute more than others to the cluster management, particularly after network deployment. Inversely, the Cost aware algorithm guarantees an $F$ value close to 1, as the workload is maximally balanced among the devices, according to their $MD$ values.

6.4 Elections

In this Section, we evaluate the total number of elections performed by smart meters to decide the current MD during the cluster lifetime. The number of
elections provides an indication of the stability of the system in terms of switching operations between Master and Slave devices. The frequency of elections impacts on the lifetime of each device $i$ through $E_{i,SD}$ and $E_{i,MD}$ values close to the worst case scenario, modeled by $E_{SD}$ and $E_{MD}$, respectively. In Figure 6(b) we plot the average number of elections (occurring from network setup till the goal is reached or the instant of the first death), while varying the number of devices in the scenario. We consider a fixed goal, equal to 10000. As before, we do not depict the No election algorithm, since in this case no elections occur at all. It is easy to see that the Highest first algorithm involves the highest number of elections, since they occur every day, unless a smart meters runs out of battery before the goal (in this case, the number of elections is equal to the lifetime of the cluster, otherwise it is equal to the goal). For the Greedy algorithm, the number of elections is at maximum the number of devices, if the system is satisfiable. Otherwise, after all the devices $i = 0, ..., n - 1$ have performed $MD_i$ days as MD, elections are issued once per day, and this explains the decreasing trend of the curve as the number of devices increases. Finally, Figure 6(b) shows that for the Cost aware algorithm the number of elections is below the number of devices, since when the system is not satisfiable each MD tries to maximize its lifetime.

7 Conclusions and Future Works

In this paper we have investigated the application of cognitive radio technologies over TV White spaces for smart metering scenarios, to overcome the current limitations of cellular technologies. To this aim, we have proposed an analytical model to study the energy consumption of a cluster of smart meters, by taking into account both the consumption for TVWS detection and for transmitting the readings to the central aggregator. Using this model, we have studied the Master/Slave scheduling problem according to recent regulations by FCC and Ofcom, and we have evaluated centralized and distributed algorithms to maximize the cluster lifetime while guaranteeing protection of the incumbents. Extensive simulations have been performed, highlighting benefits and counterfeits of the proposed solutions, in terms of lifetime maximization, network overhead and fairness of the system. Future works include: the extension of the analytical model by including power control and interference issues, the analysis of multi-hop solutions over large scale deployments of smart meters, and further comparisons of scheduling algorithms.

8 Acknowledgment

This work is supported by Italian MIUR funds under the project PRIN-2009 STEM-Net: STEM devices for self-organizing wireless NETworks.
References


Figure 5: In Figure 5(a) and in Figure 5(b) we show the percentage of goal satisfied with 10 and 500 devices, respectively.
Figure 6: In Figure 6(a) we plot the fairness of the system with a fixed goal and varying the number of smart meters, while in Figure 6(b) we see the number of elections as a function of the number of devices.