



HAL
open science

Toward heterogeneity-aware device-to-device data dissemination over Wi-Fi networks

Lyes Hamidouche, Sébastien Monnet, Pierre Sens, Dimitri Refauvelet

► To cite this version:

Lyes Hamidouche, Sébastien Monnet, Pierre Sens, Dimitri Refauvelet. Toward heterogeneity-aware device-to-device data dissemination over Wi-Fi networks. ICPADS 2017 - International Conference on Parallel and Distributed Systems, Dec 2017, Shenzhen, China. hal-01619216

HAL Id: hal-01619216

<https://hal.univ-smb.fr/hal-01619216>

Submitted on 15 Nov 2017

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.

Toward heterogeneity-aware device-to-device data dissemination over Wi-Fi networks

Lyes Hamidouche^{*‡}, Sébastien Monnet[†], Pierre Sens^{*} and Dimitri Refauvelet[‡]

^{*} Sorbonne Universités, UPMC Univ Paris 06, CNRS, Inria, LIP6

4 Place Jussieu, 75005 Paris, France

Email: Firstname.Lastname@lip6.fr

[†]LISTIC/Université Savoie Mont Blanc

5 chemin de bellevue, 74944 Annecy le Vieux

Email: Sebastien.Monnet@univ-smb.fr

[‡] Magency

60, rue de Wattignies, Paris 75012, France

Email: firstname@magency.fr

Abstract—In the last few years, there has been an explosive growth of the number of mobile devices. This has come with a plethora of new applications and usages. Among these new usages, there are many occasions for which a content has to be disseminated to a large number of mobile devices (e.g., large-scale events providing a multi-media support, video streaming, ...). To cope with network bandwidth limitations, new approaches, leveraging device-to-device (D2D) communications have emerged. Obviously, one of the main problem that D2D-based approaches have to face is the energy consumption. Furthermore, there is usually a huge heterogeneity among the devices: some may benefit of a good, fully charged battery while others may have only a couple of hours left before a power failure; the network bandwidth can also differ a lot. In this paper, based on a previous work, we propose an approach to take into account devices heterogeneity while disseminating data using D2D communications. Our simulations show that it is possible to spare the weakest batteries without wearing too much the good ones nor degrading too much the performance. Furthermore, taking into account the devices bandwidth capabilities can help to increase the dissemination speed.

Index Terms— device-to-device, efficient data dissemination, Wi-Fi Direct, heterogeneity, energy efficiency

I. INTRODUCTION

Mobile devices, such as smartphones or tablets have become ubiquitous. Their computing, storage and networking capacities are improving fast. A direct consequence is that mobile applications need an increasing amount of battery and network bandwidth. The data assets used by mobile applications are constantly increasing in size and quality. Even if the storage problem of these devices is no longer a big concern thanks to the increase of storage capacities and the possibility to extend it through removable devices such as SD cards and USB drives, disseminating voluminous data through wireless networks remains a complex task. We focus on the particular case where mobile devices are used by large-scale events participants. In this case, the devices have to be feeded with data at the beginning of the event (technical documentations, presentations, videos, ...).

To circumvent the bandwidth capacity limit, we have proposed an approach relying on D2D communications to speedup the transfers [1]. Several others works in the literature also use technologies such as Wi-Fi Direct and/or Bluetooth to tackle wireless infrastructures scalability issues. An important part of the research about the use of these communication focuses on data offloading [2], [3], [4], [5] or video streaming [6], [7], [8].

However, even if the global energy consumption is often taken into account [9], [10], [11], individual battery levels are seldom taken into account. Nevertheless, if we look at a large set of mobile devices, e.g. during a large-scale event, we observe devices with different performances, networking capabilities, and remaining battery capacity. While D2D connections are very beneficial for QoS scalability and network performance, their use is greedy in terms of energy.

In our previous work, we have paid attention to limit the global amount of energy consumed. For instance, we have made the pragmatic choice to keep a centralized orchestrator permitting to avoid many control message exchanges among devices. Our goal in this paper is not to reduce the global battery use anymore; instead, our first contribution is to preserve the batteries having the shortest remaining capacity, in order to try to let a minimal level of remaining battery after the data dissemination.

The device heterogeneity is also a matter of bandwidth. Our second contribution is to take advantage of the good bandwidth of some devices to speedup data dissemination. Obviously, our two contributions are linked: our approach will not overuse a device having a good bandwidth, it will even be preserved if its battery reaches a given threshold. Luckily, there is a good correlation: devices having a good battery are often the most recent ones, and also have good networking capacities.

This paper is structured as follows. Section II recalls the background, describes our previous contribution and related works. Then, Section III presents the model of our system and the algorithms that we propose. Section IV gives an evaluation of our approach for data dissemination, and finally, section V

concludes.

II. RELATED WORK

D2D technologies constitute an interesting candidate to tackle the performance and scalability issues of wireless networks. In fact, collaborative networking-based solutions help to increase a network's overall throughput and thus reduce the durations of data transfers. Traditional P2P approaches such as Bittorrent [12] or Gnutella [13] that have been initially introduced to solve scalability issues in wired networks do not consider some aspects of mobile devices such as their limited power, and radio interferences.

The emergence of D2D technologies such as Bluetooth and Wi-Fi Direct brings autonomous networking capabilities to mobile devices. The works in the literature tackle different issues related to D2D communications, such as infrastructures offloading [2], [3], [4], [5], opportunistic data dissemination [6], [7], [8], [14], [15], resource allocation [16] or distributed caching [17], [18]. Asadi et al. [19] have provided a very complete and detailed survey on this topic.

Before using D2D communications, it is usually necessary to perform a set-up phase. Cherif et. al [20] have investigated the problem of group formation in Wi-Fi Direct, their solution relies on electing backup Group Owners (GO) in order to speed up the re-creation of the group when the current GO disappears. In the same axis, Li et al. [21] proposed to use a dedicated common channels to enhance the discovery of communication groups in D2D networks.

The advents in wireless networks permit the simultaneous use of multiple channels to increase their communication efficiency. The use of this mode of operation is getting more and more popular and will face new challenges such as spectrum efficiency, load balancing between licensed and unlicensed channels... Naslcheraghi et al. [22] rely on full duplex enabled devices to increase data throughput in the network their simulations have shown that the devices have important gains in high intra and inter cluster interferences. Lee et al. [16] focuses on bringing the newly arrived devices of a D2D network the most suitable channels for D2D networks. With the churn that a mobile network can sustain, it is important to always provide efficient channels for newly arrived devices. Their online algorithm was able to converge quickly and performs with a performance close to the optimal that an offline scheduling algorithm with prior information of the arrival of mobile devices.

Concerning data distribution, Pan et al. [17] have pointed the lack of efficiency of broadcasting through Wi-Fi Direct groups. They have proposed data dissemination models based on Wi-Fi Direct API model and compared a ring-based and tree-based multi-group Wi-Fi Direct network.

Closer to our works, MicroCast [8] and MoVi/MoVi+ [6], [7] simultaneously use server downlink and direct communications among mobile devices to improve on-demand video. They both rely on using a unique downlink with the server that hosts the information (video data) and use their ability to establish spontaneous D2D communications to exchange data.

Keller et al [8] (MicroCast) propose on a set of mechanisms that provides the ability to improve data retrieval among devices by relying on the broadcast nature of Wi-Fi networks. Adapting such kind of solution for large scale networks would require an important control to make sure all the packets are received by devices. Furthermore, broadcasting will provoke many retransmissions due to network interferences, which slows down data dissemination.

The authors of [6] and [7] use adhoc communications between devices to improve network QoS for collaborative on-demand video. They rely on a centralized system architecture to achieve their performance goals. In fact, they use centralized infrastructure devices as supervisors for the state of the network and the interferences fluctuations to opportunistically trigger exchanges between devices. In our approach, the fluctuations of wireless interferences help to define a limit of mobile devices that communicate simultaneously in the network. This simulation degree can vary depending on network environment conditions. In their work, the authors have also focused on improving the switch between adhoc and infrastructure Wi-Fi. Using Wi-Fi Direct for D2D communications between devices relies on the same mechanisms with infrastructure Wi-Fi and current commercial devices are capable of handling simultaneously a communication group and an infrastructure link with access points. By using this mechanism, we are able to establish D2D connections only while needed and adapt the parallelism degree in the network.

Concerning power management, it is a very important aspect in mobile networking and D2D communications make no exception. In fact, exploiting D2D communications to improve infrastructure QoS helps to reduce overall networking time. However, it has an important energy cost. Globally reducing network overall power consumption is important, however, by not considering the state of the individual mobile devices that participate to the network, some devices can be over-used in comparison with their neighbors. Furthermore, in the current mobile devices ecosystem, we have a considerable heterogeneity of mobile devices. Their characteristics (Battery life, computing capacity, bandwidth, ...) vary depending on their brand, their model, Moreover, several devices can have a half charged battery. To the best of our knowledge, most of the contributions around power efficiency in D2D networks tackle the problem with the goal of reducing the network overall consumption and do not consider the heterogeneity of the mobile devices in terms of remaining battery capacity. Nobach et al. [11] treat the problem of the security and power consumption that mobile devices face while actively scanning to discover the nearby devices of the network. Power saving and content delivery efficiency are two important and conflicting aspects in D2D-based communication networks. Sung et al. [9] contribution consists in two algorithms for packet routing and scheduling in order to balance between these two factors. Their simulation for small-scale networks has shown the possibility of balancing between a longer D2D networking and a better content delivery performance.

With the heterogeneity of mobile devices, treating all the de-

vices in the same way can lead to important over-participation of the first devices to acquire data. This can lead to important battery losses among these devices. It is thus mandatory to consider the devices separately while disseminating data in a D2D based network. The devices with the weakest batteries that participate to the dissemination should be protected of any overuse in the network. This is particularly important when considering the concerns of a company that animates large scale events (such as Magency [23]), where it is mandatory to leave the devices with enough battery to last all along the event. It is even preferable to increase the overall power consumption if it provides the ability to preserve the weakest devices from being overused. However, the use this kind of approach would requires the approval of users to consume more energy to save bad devices batteries, but when the devices are lent to users (like during large scale events), it becomes simpler to implement them. In practice, when a user refuses to share his device's energy, his device is considered as a weak battery one.

III. CONTRIBUTION

In our previous work [1], we have presented EDWiN, our approach to leverage D2D communications among Wi-Fi Direct enabled devices in order to improve the network Quality of Service for data dissemination. The algorithms that we introduce here are an extension of our previous work [1] that takes into account the energy and the bandwidth of each device. In this section we first describe the architecture and the model of the considered system, providing the necessary background concerning EDWiN; then we describe the proposed algorithms.

A. Architecture

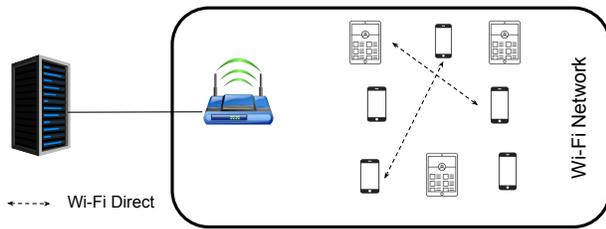


Figure 1. System representation

In our study, we focus on the the case of Wi-Fi infrastructures that are used to host large scale events. Figure 1 represents the architecture of our system. It consists in:

The server : the node hosting the data to be downloaded by mobile devices;

Wi-Fi infrastructure : it consists in routers, wireless controllers and access points. It is used to create a network that the mobile devices will connect to, as the only link with the server;

Mobile devices : the nodes accessing data, these devices are also Wi-Fi Direct enabled.

The goal is to disseminate the data hosted by the server on all the mobile devices as fast as possible. The server must be able to provide the devices the data they require and leave enough bandwidth for other applications that communicate through the same medium. We pragmatically use the server as a scheduler. The network's global state is then maintained with control messages exchanged with the clients (mobile devices). Thus, data transfers and exchanges between devices can be easily scheduled as the server has a global view, maintaining a "Network State Table" (*NST*) that holds all the information about the devices (including disseminated chunks).

System representation: Depending on the network that has to be covered by a Wi-Fi infrastructure, the number of the servers, routers and access points (APs) can vary. In our study we focus on a part of such a network. We the consider a Wi-Fi network \mathcal{N} which is composed of $|\mathcal{N}|$ mobile devices that are connected to one single server \mathcal{S} , through a single access point. The server disseminates a voluminous data to the mobile devices. This data is split into chunks in order to quickly increase the number of data sources in the network (smaller chunks will be transfered faster, new sources will be available sooner). As our goal is to offload the infrastructure, we restrict the number of concurrent downloads from the server to one. In practice this parameter can vary depending on the infrastructure and the bandwidth that has to be left for other network usages.

As D2D exchanges require a connection establishment that lasts a non negligible time, it is better for devices to exchange the maximum amount of data possible at each D2D connection.

Wireless networks interferences depend on the environment and are difficult to model or reproduce. Obviously, the more parallel exchanges there are, the more interferences there will be. Our approach is thus tunable, a limit upon the "parallelism degree" can be set. It corresponds to the maximum number of devices authorized to exchange data simultaneously in the network. We note this parameter *maxPara*.

Device representation: Every mobile device $d \in \mathcal{N}$ is defined by parameters that are used by our algorithms to schedule the exchanges between devices:

State : defines the current networking state of the device. A device can be idle or exchanging with another device or the server;

Chunks : is a set representing the chunks that have been retrieved by the device;

Battery life : in seconds, represents the remaining battery of the device;

Networking battery loss factor : represents the battery loss speed of the device when networking intensively ;

Device-to-device throughput : represents the throughput of the device when sending data through a D2D connection.

During an event, the battery of the mobile devices should last long enough to let all the participants use them all along

the day. This means that each battery should remain over a given level at the end of the dissemination. We thus set a parameter which is the minimal battery threshold under which our solution prevents a device from being overused by D2D communications. At any time the network is thus partitioned into two sets of devices that we call “Good” and “Bad” depending if the remaining battery is above, resp. below the threshold. Obviously, the partitions are dynamic: a device considered as “Good” can become “Bad” if it consumes too much power (for instance, old devices usually consume their battery faster than recent ones). We observed on a set of devices that are used during large-scale events that their battery life and battery loss speed vary depending on the age and the model of the devices¹.

The device-to-device throughput also varies a lot among devices. Devices have different Wi-Fi chips, and usually recent devices are equipped with cutting edge chips capable to handle higher data throughput than old ones. Leveraging “fast” devices can help to reduce the total dissemination time, however, it is important to pay attention to their battery usage.

In our previous work [24], we considered only homogeneous configurations. We mainly focused on the importance of parameters such as the parallelism degree and the data size and their impact on the data dissemination performance.

In the following of this paper, we focus on the battery and bandwidth throughput heterogeneity. In terms of energy usage, our goal is not to reduce the global consumption, but to prevent the device batteries to go under a given level. If the global network battery use is a little bit higher, it is not a problem as long as the devices with the worst batteries consume less energy.

Our goal remains to reduce the dissemination completion time of voluminous data over Wi-Fi networks by using parallel D2D connections between devices. Furthermore, we want to offload the networking infrastructure (Wi-Fi Access Points, routers and servers). Considering our case study, a dissemination happens generally at the beginning of a large scale event or at the update of a voluminous data.

In the remainder of this section we detail three different approaches that help to reduce data dissemination time with different considerations of the network.

B. Naive dissemination algorithms

Initially, the purpose of our algorithm is to quickly increase the number of data sources in the network and the exchanges between devices. The dissemination starts when the server sends the first chunk to a device randomly selected then the server is acknowledged by the devices at every end of transfer or exchange between devices. In order to replicate the chunks uniformly, the server uses a “rarest first” policy to select the data chunks to transfer/exchange among mobile devices. Thus, there are quickly many replicas of the chunks on

different devices, which favours the D2D exchanges among the mobile devices. In order to use D2D connections efficiently, the server selects pairs of devices which have the maximal hamming distance on their chunk-sets. In other words, the server selects the devices having the maximum amount of data to exchange. While there are available devices and the number of active devices *activeDevices* has not reached the maximum parallelization limit *maxPara*, the server schedules as much D2D exchanges as possible.

Without considering the battery life, the first devices to receive chunks will often be selected for exchanges with the other devices. This high activity may cause an important energy consumption on some devices. In the meantime, other devices use a much smaller amount of energy for exchanging (the energy consumption is not well balanced). This naive algorithm has been introduced and detailed in our previous contribution [1].

The performance in terms of dissemination time is good, however, unfortunately in practice, the high activity can occur on devices having bad or low-level batteries. In this case, they will have less chances to last for the rest of the event.

C. Battery aware D2D dissemination algorithm

Considering the battery life of the devices is important to permit to every user device to last the event. We observed, through measurements driven on devices loaned by Magency, that the throughput using Wi-Fi Direct connections was more than 10 times slower than the throughput using an access point. This is due to the connection times that are not negligible but also, and mainly because the transfers themselves are slower. A D2D network hyperactivity on a device could provoke important battery consumption and hinder the user from using its mobile device all along the event.

Algorithm 1 Battery aware dissemination policy

```

1:  $\mathcal{I} \leftarrow \{ \forall d \in \mathcal{N} ; d.state = Idle \}$ 
2:  $good \leftarrow \{ \forall d \in \mathcal{I} ; d.battery \geq minBattery \}$ 
3:  $bad \leftarrow \{ \forall d \in \mathcal{I} ; d.battery < minBattery \}$ 
4: Disseminate chunks to devices  $d \in good$  until
    $activeNodes \geq maxPara - 2$ 
5: while  $\neg disseminationFinished$  do
6:   if  $bad = \emptyset$  then
7:     do
8:       Perform naive approach
9:     until  $bad \neq \emptyset$  or  $dissertationFinished$ 
10:   else
11:     send data to  $d, \forall d \in bad$ 
12:     keep scheduling exchanges among devices  $\in good$ 
13:   end if
14: end while

```

In order to protect the devices having bad or low-level batteries, we do not use them for D2D exchanges. Furthermore, when a bad device downloads data from the server, it

¹observations made on the devices provided by Magency [23] for their customers

downloads all the data in one time to limit the number of connections/disconnections.

In order to have enough parallel transfers between devices, the server disseminates enough chunks to keep as many devices active in parallel as the parallelism degree ($maxPara$) allows. Then the server sends data to devices with weak batteries, sequentially. If a node which was first considered as good becomes bad (because it has use too much its battery) it will enter the bad set, and the missing data will be downloaded directly from the server. If the bad set becomes empty, then the server helps the good devices, as in the naive approach. Algorithm 1 describes how the server handles the transfers in the network.

If there never bad devices in the network, the server simply acts with the same behaviour as the naive algorithm.

D. Battery and bandwidth aware approach

The heterogeneity does not only come from battery levels. Some devices have more recent Wi-Fi chips and are able to handle higher data throughput. Taking into consideration this parameter provides the ability to improve the network performance and reduce the global dissemination time. Giving priority to several devices having a large bandwidth could help to quickly obtain fast data sources and thus, to speed up the global dissemination. These particular data sources (several devices having both a good remaining battery and a large bandwidth) can thus be used to feed the other devices.

We propose an approach that takes into account both battery and bandwidth heterogeneity. We do not assume that devices with a good bandwidth are also those having a good battery. However, in practice, recent devices usually have a better battery and a larger bandwidth. These two parameters seems correlated. In our approach presented below, we rely on devices having both a good battery and a large bandwidth (when they exist in the system).

Algorithm 2, executed at the beginning and at the end of each transfer, describes the server’s scheduling policy. Inside the good devices set, it considers devices based on their bandwidth throughput. We call “Fast”, the devices that have a data throughput greater that a defined limit $minFastBW$. Fast devices can exchange quicker data inside their set than with other devices. From line 1 to 5, the algorithm computes the different sets of devices based on the idle devices characteristics.

Then, if there are currently no transfer from the server in progress ($AP.state = Idle$, Algorithm 2 line 6), the server triggers a transfer through the AP by calling the function $scheduleAPTtransfer()$.

This function’s behavior depends on the state of the network. If there are not enough fast data sources (the fast devices), the server prioritizes the data dissemination on these devices. Then, if there are no good devices available, the server disseminates data to bad devices.

Concerning D2D exchanges, the server selects fast devices that have retrieved all the data as data sources for other good devices with lower throughput and finally to bad devices

Algorithm 2 Battery and bandwidth aware dissemination algorithm

```

1:  $\mathcal{I} \leftarrow \{ \forall d \in \mathcal{N} ; d.state = Idle \}$ 
2:  $good \leftarrow \{ \forall d \in \mathcal{I} ; d.battery \geq minBattery \}$ 
3:  $bad \leftarrow \{ \forall d \in \mathcal{I} ; d.battery < minBattery \}$ 
4:  $fast \leftarrow \{ \forall d \in good ; d.bandwidth \geq minFastBW \}$ 
5:  $fastCompleted \leftarrow \{ \forall d \in fast ; d.completed = True \}$ 
6: if  $AP.state = Idle$  then
7:    $scheduleAPTtransfer()$   $\triangleright$  Transfer from the server
8:    $update()$ 
9: end if
10:  $p \leftarrow selectExchange()$ 
11: while  $p \neq Nil$  and  $activeDevices < MaxPara - 2$  do
12:   Trigger exchange between  $p.device1$  and  $p.device2$ 
13:    $activeDevices \leftarrow activeDevices + 2$ 
14:    $update()$   $\triangleright$  Updates the lists above
15:    $p \leftarrow selectExchange()$ 
16: end while
17: function  $scheduleAPTtransfer()$ 
18:    $d \leftarrow Nil$ 
19:   if  $|fastCompleted| < N_{fast}.percent$  then
20:      $d \leftarrow selectFastDevice(fast)$ 
21:   else if  $good \neq \emptyset$  then  $\triangleright$  Good but not fast
22:      $d \leftarrow selectDevice(good)$ 
23:   else
24:      $d \leftarrow selectDevice(bad)$ 
25:   end if  $\triangleright$  Bad battery and slow
26:   if  $d \neq Nil$  then
27:      $data \leftarrow chunksForDevice(d)$ 
28:      $Send(< data, d >)$ 
29:   end if
30: end function

```

in order to disseminate chunks in one connection. It is a way to prevent the bad devices from multiple connection establishments that could waste their energy. Note that the devices having a bad/low battery will get filled at the end. This allows the server to estimate the time they’ll have to wait before getting served, permitting these fragile devices to go into sleep mode.

If the devices having both a good battery and a good bandwidth have already retrieved all the data, the server schedules exchanges between devices by prioritizing fast transfers as described by Algorithm 3. When no such good device is available, the algorithm falls back on the naive approach, based on the hamming distance computed on the chunk sets ($selectHamming()$).

IV. EXPERIMENTATION AND EVALUATION

A. Evaluation conditions

We evaluated our approach using Peersim [25], a discrete event simulator. In order to calibrate our simulator, we have run extended measurements on hardware loaned by Magency [23]. This experiment campaign has allowed us to observe the behavior of a Wi-Fi infrastructure centralized on

Algorithm 3 Exchange selection function

```
1: function selectExchange()
2:   if  $|fastCompleted| > 0$  then
3:      $fast \leftarrow fastCompleted.pop()$ 
4:     if  $|good| > 0$  then
5:        $d \leftarrow selectNode(good)$ 
6:       return ( $fast, d$ )
7:     else if  $|bad| > 0$  then
8:        $d \leftarrow selectNode(bad)$ 
9:       return ( $fast, d$ )
10:    end if
11:  else
12:    return selectHamming( $good$ )
13:  end if
14: end function
```

a single access point; to measure Wi-Fi Direct connection establishment times, unidirectional and bidirectional D2D throughputs, battery consumption under various conditions. Our prototype is composed of two layers: (i) an applicative layer and (ii) a transport layer. The applicative layer implements the different exchanges between devices. The transport layer, which we calibrate with the preliminary measures, permits to compute latencies and bandwidth between pair of nodes.

For the evaluation of our approach, we simulate a network of 200 mobile devices that have to retrieve a 500MBytes data block. This data is split into 10 chunks (50MBytes each). Only 33 devices are authorized to exchange simultaneously in the network ($maxPara$)². The devices download from the server (AP) at a 10MBytes/second data rate. For D2D transfers, the data rate varies depending on the device that is considered. We have observed, that depending on the brand of a device, or its operating system, the data rate of D2D connections vary. For instance, we have observed practical data rate of 1Mbyte/s between a pair of our Apple iPad Air Tablets, whereas, with a Samsung Galaxy S7, we have observed a 4Mbytes/s data rate. We use these observed data rates as values to define slow and fast devices. Based on devices that are distributed to the attendees of corporate events, we observed that the battery life depends on their age and condition. The good devices we simulate here have a 12 hours battery life and the bad devices have a 8 hours battery life which is also the battery level threshold under which a device gets “protected” by our approach ($minBattery$).

In terms of energy saving, our goal is not to reduce the overall network consumption. Instead, we try to preserve the devices with the weakest batteries from important battery losses. In terms of bandwidth heterogeneity, our goal is to reduce dissemination time, by quickly disseminating all the data to fast devices. Then the approach rely on these nodes to transfer the data to other device using one single connection, reducing the number of connection establishments.

We have chosen to not simulate the network interference because it depends on the environment and it is hard to model

correctly, and even real experiments in presence of interference are hardly reproducible. It is thus important to notice that our simulator gives results considering an ideal, interference-free environment. However, as detailed in [1] our approach provides the ability to vary the parallelization degree in order to lower the interference level.

The performance metrics that we evaluate in our study are the following.

- **The completion time** : the time it takes to disseminate the data to all the devices. It represent the performance of the dissemination. We have observed that using the naive approach most of the nodes get completed close to the completion time (in a short time frame).
- **The estimated remaining battery** : the estimated remaining battery at the end of the dissemination.

We first evaluate the ability of our approach to preserve a minimum amount of battery on fragile devices (to allow its user to attend the event). We also evaluate the impact of the proportion of devices having a bad battery, or a large bandwidth on the completion time. Our goal being to preserve the bad batteries while still offering a good performance in terms of completion time.

B. Considering battery duration

We remind here that the objective is not to reduce the network global energy usage but to preserve low level batteries. Figure 2 shows the remaining battery difference in comparison with the naive approach for various percentages of bad devices in the network. We measure here the average estimated remaining battery in the network, the mean remaining battery for the “bad” devices and the worst device battery. We observe, that for the worst devices the battery-aware approach saves up to 2 hours of battery on the bad devices in comparison versus the naive approach. We also observe that the global battery use in the network is better with the battery aware approach, except for 40% of bad devices in the network. This is thanks to the fact that in the battery-aware approach, “bad” devices do not need to create D2D links with other devices, there are less connections/disconnections, saving networking time, and thus battery.

Figure 3 shows the impact of the percentage of bad devices on the completion time. There again, we compare the performance of the naive dissemination approach that considers all the devices in the same way versus the battery aware approach. We observe that under 30% of bad devices in the network, the battery aware approach outperforms the naive approach. As in the naive approach we already limit the number of parallel transfers, there is no more parallel transfers than in the battery-aware approach; furthermore, in the battery-aware approach, we save some connection/disconnection time as mentioned above.

However, for a higher proportion of bad devices, we can observe a performance loss. In fact, as we can see in Figure 4; which shows the evolution of the number of completed node in both good and bad device sets and the whole network; all the good devices retrieve the whole data quickly while the bad

²More explanations on the parallelization limit can be found in [1].

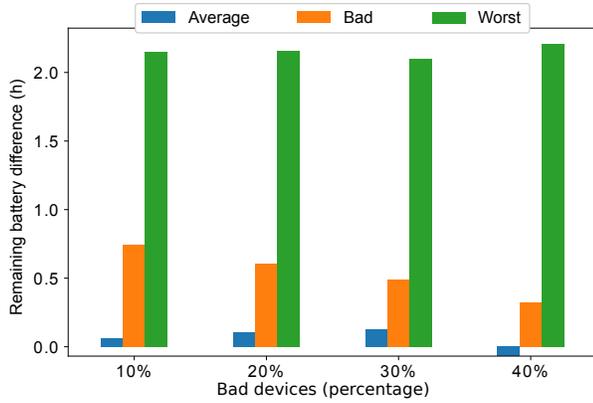


Figure 2. Remaining battery difference with battery aware approach in comparison with the naive approach

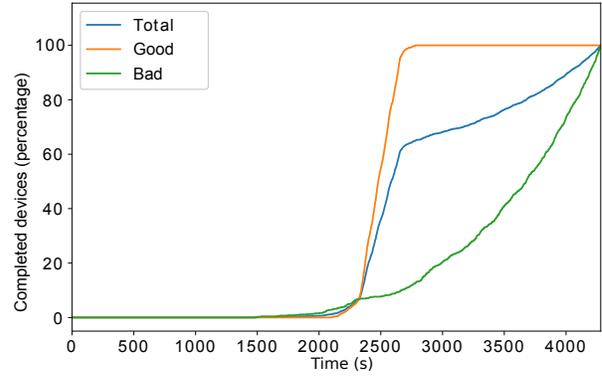


Figure 4. Completion evolution with battery aware approach with 40% of bad devices in the network.

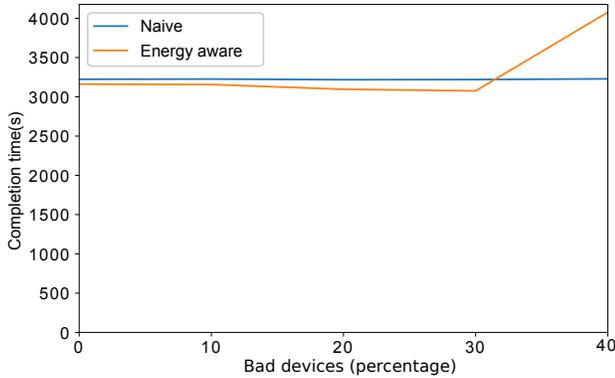


Figure 3. Completion time comparison between naive and battery aware approach - (By %-age of bad devices)

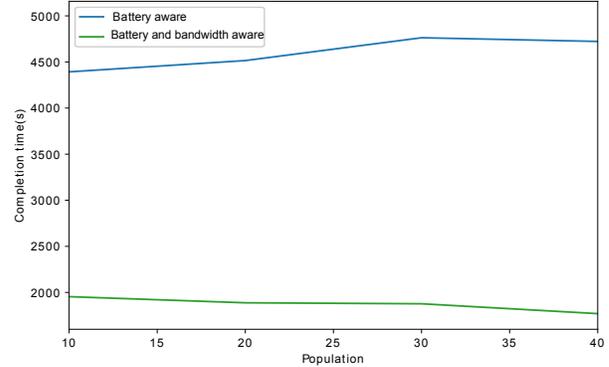


Figure 5. Performance of Bandwidth aware approach in comparison with bandwidth aware approach

devices are still downloading from the server. When there are too many devices with a bad battery the sequential download by these devices as a strong impact on the completion time.

Using this approach might prevent from important battery losses of the worst devices, but it can cause a performance drop of the dissemination when the proportion of device having a low-level battery is too high.

C. Considering battery duration and bandwidth

Figure 5 show the completion time while varying the percentage of fast devices (device with high bandwidth) in the network. In these simulations, there are 40% of bad devices (devices with a low-level battery). We measure the impact of the percentage of fast devices on the performance of the network. We observe that, by using a hybrid approach we have better completion time than with the battery-aware approach only: it is worth taking bandwidth heterogeneity into account. We also observe, that, as expected, increasing the proportion of those devices in the network improves the performance of the dissemination.

Figure 6 shows the percentage of battery gain we obtain when using the hybrid approach in comparison with the

battery-aware approach. We observe that we have a better energy gain. By considering the use of fast devices in the dissemination, data sources with the whole data appear quickly in the network. The other mobile devices receive the whole data in one connection from another devices. Leaving the bad devices as the last to get data from the server permits to save their energy and retrieve data at the end of the dissemination. We also observe that the energy gain increases with the increase of the proportion of fast devices. In fact, thanks to a higher number of fast D2D nodes in the network, the devices with good batteries retrieve the whole data faster, the devices with weak batteries wait less to get data. Thus, devices spend less time networking and save energy.

To put it in a nutshell, preventing devices with bad batteries from any exchange with other devices can be efficient, but only when their proportion is small. It can even provide a performance gain thanks to a lower number of connections/disconnections. When their proportion is too big, the sequential transfers from the server to the bad devices take too much time, even after the good devices have finished their exchanges, end are completed.

If the devices bandwidth are heterogeneous, it is worth

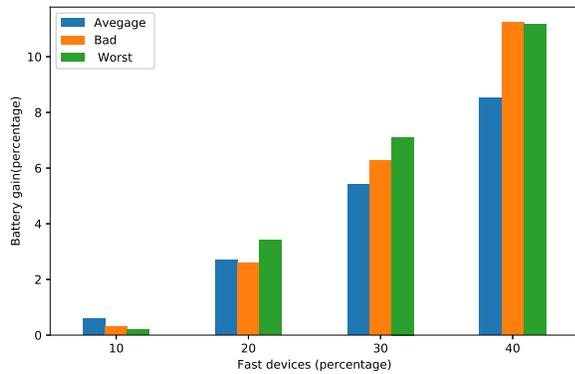


Figure 6. Battery gain comparison between the energy aware approach and the hybrid (Energy + bandwidth) approach

taking this into account. As expected, this provides the ability to save dissemination time (performance gain), but also energy (a shorter dissemination consumes less energy).

V. CONCLUSIONS AND FUTURE WORK

In this paper, we focused on the impact of devices heterogeneity in a D2D based data dissemination system. We proposed two dissemination approaches: one that considers the case where the devices differ only in terms of battery and one that considers an heterogeneity in terms of both battery and network bandwidth.

To prevent an overuse of the devices with weak batteries we prevent them from participating to the energy-consuming D2D exchanges. We also leverage devices having both a good battery and a good Wi-Fi chip to enhance the dissemination performance while still preserving low-battery devices.

We have discussed in our evaluations how performance could be impacted by the proportion of devices which batteries should be preserved. We have shown that when the proportion is small it is possible to preserve the bad batteries with no performance degradation, but a small performance gain. Furthermore taking into account network bandwidth heterogeneity can enhance both the completion time and the energy saving.

In the future, we plan to investigate how the network bandwidth heterogeneity can be used to disseminate data from a device with high data throughput to several devices with smaller bandwidth. We also plan to study other dissemination strategies, using hierarchical topologies.

REFERENCES

- [1] L. Hamidouche, S. Monnet, F. Bardolle, P. Sens, and D. Refauvelet, "Edwin: Leveraging device-to-device communications for efficient data dissemination over wi-fi networks," in *2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA)*, March 2017, pp. 228–235.
- [2] S. Dimatteo, P. Hui, B. Han, and V. O. K. Li, "Cellular traffic offloading through WiFi networks," *Proceedings - 8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems, MASS 2011*, pp. 192–201, 2011.
- [3] F. Rebecchi, "Device-to-device data offloading: from model to implementation," Ph.D. dissertation, Université Pierre et Marie Curie-Paris VI, 2015.

- [4] S. Andreev, A. Pyattaev, K. Johnsson, O. Galinina, and Y. Koucheryavy, "Cellular traffic offloading onto network-assisted device-to-device connections," *IEEE Communications Magazine*, vol. 52, no. 4, pp. 20–31, 2014.
- [5] A. Pyattaev, K. Johnsson, S. Andreev, and Y. Koucheryavy, "Proximity-based data offloading via network assisted device-to-device communications," in *Vehicular Technology Conference (VTC Spring), 2013 IEEE 77th*. IEEE, 2013, pp. 1–5.
- [6] H. Yoon, J. Kim, F. Tan, and R. Hsieh, "On-demand video streaming in mobile opportunistic networks," in *Pervasive Computing and Communications, 2008. PerCom 2008. Sixth Annual IEEE International Conference on*. IEEE, 2008, pp. 80–89.
- [7] H. Lee, J.-Y. Yoo, and J. Kim, "Movi+: Opportunity extension for mobile peer-to-peer video on demand," in *Consumer Communications and Networking Conference (CCNC), 2013 IEEE*. IEEE, 2013, pp. 247–252.
- [8] L. Keller, C. Fragouli, and U. C. Irvine, "MicroCast : Cooperative Video Streaming on Smartphones Categories and Subject Descriptors," *MobiSys 2012*, pp. 57–69, 2012.
- [9] J. Sung, D. Lee, Y. Bang, J. Lee, and J. K. K. Rhee, "Energy-aware algorithms for network-assisted device-to-device content delivery networks," in *2015 International Conference on Information and Communication Technology Convergence (ICTC)*, Oct 2015, pp. 469–471.
- [10] F. Wang, C. Xu, L. Song, Q. Zhao, X. Wang, and Z. Han, "Energy-aware resource allocation for device-to-device underlay communication," in *2013 IEEE International Conference on Communications (ICC)*, June 2013, pp. 6076–6080.
- [11] L. Nobach and D. Hausheer, *Towards Decentralized, Energy- and Privacy-Aware Device-to-Device Content Delivery*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 128–132.
- [12] R. Toole, "BitTorrent Architecture and Protocol," 2006.
- [13] M. Ripeanu, "Peer-to-peer architecture case study: Gnutella network," *Proceedings First International Conference on Peer-to-Peer Computing*, pp. 1–11, 2001.
- [14] N. Golrezaei, M. Ji, A. F. Molisch, A. G. Dimakis, and G. Caire, "Device-to-device communications for wireless video delivery," *2012 Conference Record of the Forty Sixth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, no. March, pp. 930–933, 2012. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6489152>
- [15] Q. Gong, Y. Guo, Y. Chen, Y. Liu, and F. Xie, "Design and evaluation of a wifi-direct based lte cooperative video streaming system," in *Global Communications Conference (GLOBECOM), 2016 IEEE*. IEEE, 2016, pp. 1–6.
- [16] G. Lee, W. Saad, M. Bennis, A. Mehdodniya, and F. Adachi, "Online channel allocation for full-duplex device-to-device communications," in *Globecom Workshops (GC Wkshps), 2016 IEEE*. IEEE, 2016, pp. 1–6.
- [17] M.-S. Pan and Y.-P. Lin, "Efficient data dissemination for wi-fi peer-to-peer networks by unicasting among wi-fi p2p groups," *Wireless Networks*, pp. 1–19.
- [18] B. Chen and C. Yang, "Caching policy optimization for d2d communications by learning user preference," *arXiv preprint arXiv:1704.04860*, 2017.
- [19] A. Asadi, Q. Wang, and V. Mancuso, "A survey on device-to-device communication in cellular networks," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 4, pp. 1801–1819, 2014.
- [20] W. Cherif, M. A. Khan, F. Filali, S. Sharafeddine, and Z. Dawy, "P2p group formation enhancement for opportunistic networks with wi-fi direct," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, March 2017, pp. 1–6.
- [21] H. B. Li, R. Miura, and F. Kojima, "Channel access proposal for enabling quick discovery for d2d wireless networks," in *2017 International Conference on Computing, Networking and Communications (ICNC)*, Jan 2017, pp. 1012–1016.
- [22] M. Naslcheraghi, S. A. Ghorashi, and M. Shikh-Bahaei, "Full-duplex device-to-device collaboration for low-latency wireless video distribution," *arXiv preprint arXiv:1704.03704*, 2017.
- [23] "Magency," <http://www.magency.fr/>.
- [24] L. Hamidouche, S. Monnet, F. Bardolle, D. Refauvelet, and P. Sens, "EDWiN: leveraging device-to-device communications for efficient data dissemination over Wi-Fi networks [to appear]," Taipei, Taiwan, Mar. 2017.
- [25] "Peersim," <http://peersim.sourceforge.net/>.