

SPECIAL ISSUE PAPER

Context-awareness over transient cloud in D2D networks: energy performance analysis and evaluation

Igor Bisio¹, Fabio Lavagetto¹, Andrea Sciarrone^{1*}, Terrence Penner² and Mina Guirguis²¹ University of Genoa, Genova, Italy² Texas State University, San Marcos, TX, USA

ABSTRACT

The exponential increase in the number and types of mobile devices, along with their ever-growing sets of capabilities, have enabled the development of new architectures that aim to harness such heterogeneity. Transient clouds are examples of mobile clouds, which are created on-the-fly by the devices present in an environment to share their physical resources (e.g. CPU, memory and network) and would disappear as the nodes leave the network. They enable a device to go beyond its own physical limitations through utilising the capabilities offered by nearby devices over an ad hoc network. This idea exploits the device-to-device (D2D) communications paradigm, which allows two nearby devices to communicate with each other in the licenced cellular bandwidth without a base station involved.

In this paper, we present a transient context-aware cloud (TCAC) paradigm based on the assumption that the nodes of the network care more about providing/learning higher level functionalities rather than lower level capabilities in D2D scenario. The proposed architecture, realised by using a WiFi Direct, can be portable through any paradigm, which exploits the D2D communications, so opening the doors to forthcoming 5G scenarios.

We present a prototype implementation of our architecture over Android smartphones connected via WiFi Direct along with the performance metrics (power/energy consumption and accuracy) to show the benefits of TCAC. A theoretical and analytical model for the energy consumption related to a device within the TCAC is provided as well. Copyright © 2015 John Wiley & Sons, Ltd.

*Correspondence

A. Sciarrone, University of Genoa, Via dell'Opera Pia 13, 16145 Genoa, Italy.

E-mail: andrea.sciarrone@unige.it

Received 29 June 2015; Revised 6 October 2015; Accepted 20 October 2015

1. INTRODUCTION

Motivation: Despite the recent technological advances in mobile devices, a single device has many limitations in terms of the amount of computation that can be done locally and the extent to which power is consumed. This has prompted a lot of research efforts in the areas of cloud computing and mobile cloud computing in which computations are offloaded to the cloud [1, 2]. In some scenarios, however, the cloud is ill-suited to carry out the computation due to its lack of context – something that only the devices present in the environment can capture. It would be costly to have all the devices provide the cloud with that context. Moreover, delay and privacy constraints exacerbate such issues. Consider the case in which a computation is needed for a real-time context query: by the time the cloud service is invoked, the context has already changed. Similarly, performing the computation on the cloud (where auditing and

monitoring tools are more prevalent) may not provide some users with the desired degree of privacy.

Understanding the local context is important in many applications. For example, in a health monitoring service setup it is important to recognise certain activities (e.g. Is the person moving or not? Does this activity pattern resemble a known one? Does he/she sound distressed?, etc) or lack of activity and raise an alarm if the recognised activity does not match the expected one. The context should be tied to the time of day and the known patterns for that individual. Similar examples abound in other scenarios as well. As such, all the relevant information pertaining to the context must be considered – including the users and the applications themselves [3]. Often, such information spans more than one device due to the inherent difference in capabilities supported by each device. Fusing such information for context awareness is our main motivation behind this work.

Clearly, context awareness requires some computation and communication among the present devices. The transient cloud model allows for devices to rely on their collaborative efforts to provide computational services without access to the infrastructure [4]. To that end, this paper presents an architecture that fuses a context awareness algorithm (inspired by [5]) with transient clouds ([4]). Rather than focus on sharing physical resources such as the CPU, memory, storage and network, we focus on sharing higher level functionalities that are driven by the context. We argue that in many scenarios, devices are more interested in the context rather than the exact computation to be performed on a remote device. Furthermore, our proposed architecture does not require a connection to the infrastructure, which may not always be available (e.g. disaster impacted sites and monitoring in remote areas) or usable in some situations (e.g. high traffic load that congest the network links).

Illustrative examples: Consider a major traffic congestion incident in which all the drivers – through their smartphones – are trying to compute alternative routes. Despite the fact that all drivers are going towards different destinations, getting out of this congested region starts with a much smaller set of routes. It would be worthwhile for only a few of the devices to download the new maps, compute the alternate route and share the result (i.e. the context) with other drivers rather than have all the drivers try to download the maps – creating network congestion – and perform almost the same exact computation themselves.

In another example, consider a private meeting among a group of users in which the goal is to infer and share high-level information about the meeting such as how many people were attending, who was speaking and for how long. With context-aware services, devices can detect that the meeting is ongoing and, based on their capabilities (and also proximity to certain users), some can start recording while other devices can run feature selection and identification to extract who spoke and for how long. Such information can be shared with all the users after the meeting has concluded (e.g. meeting min). This can be realised without sending information over the traditional cloud service, because a high degree of privacy may be necessary.

One of the main contributions of this work is merging the concept of transient clouds with context awareness and demonstrating the usefulness of this approach through experimental evaluation with a heterogeneous set of devices in realistic applications. The idea, backed up by results, allows obtaining a significant energy saving when several mobile devices are employed. In this paper, we present a practical application of transient clouds with context awareness for the aim of inferring some of the main speakers' characteristics (e.g. gender, identity and language) from his voice signal. Furthermore, we focus on energy consumption for analysing and measuring the effectiveness of our platform.

Paper organization: In Section 2, we describe various pieces of work related to our proposed platform. In Section 3, a brief introduction of the device-to-device (D2D) communication and its employment to the proposed architecture is provided. In Section 4, we describe our proposed platform in detail. Section 5 concerns an analytical model, which represents the energy necessary to offload some computation to another mobile device within the transient context-aware cloud (TCAC). We present our results in Section 6, and we conclude the paper in Section 7 with a summary.

2. RELATED WORK

The continuous development of hardware sensors on mobile devices has enabled the devices to pervasively recognise the social context of their environment. Consequently, understanding daily occurring human-centric actions, activities and interactions by using a smartphone has drawn much interest from the research community in the context-awareness area. In this section, we give an overview of some of the work on the topic of context-awareness followed by an overview of the research work in the area of mobile clouds.

In context awareness, the authors in [6] state that understanding the context means inferring the situation of a person, a place or an object. Any information from the interaction between a user and an application that can be used to characterize this situation is considered relevant. This includes the users and applications themselves. In the literature, many works tackle the problem of inferring high-level information from smartphones (see, among others [7–10] and references therein). A first classification of context-aware algorithms can be effectuated by considering the specific signal they work with. In particular, [7] proposes an activity recognition method designed to distinguish four different user activities by periodically classifying accelerometer signal frames using a decision tree approach. The work in [10] describes a new location recognition algorithm for automatic check-in applications, suited to be implemented over smartphones. The algorithm uses GPS and HPS positioning information together with data received by WiFi to validate the users check-ins. Concerning the audio signal, [11] and [5] propose algorithms to recognise a speaker's characteristics, such as gender, identity, language, and emotional state, by using support vector machines as classifiers.

In mobile cloud computing, there has been much research that has focused on offloading computation to reduce the toll on smartphones. The works in [2] and [12], among others, offload computation from a smartphone to a remote server in the public cloud. The main drawback of the aforementioned papers is that they rely on a working Internet connection, which may not be available or feasible in many real scenarios. Mobile clouds aim to tackle this issue by relying on the devices themselves [13]. The work in [14, 15] show various examples of mobile clouds with the goal to balance the load on devices while taking

dependencies between different tasks in consideration. Some works look at how to schedule tasks in the traditional cloud [16] or the network infrastructure [17] instead of the devices themselves. A related field of study is that of sharing information in vehicular networks, like in [18]. Our work in ad hoc mobile networks goes farther than that of [19] by passing around not just text messages but other forms of context relevant to the devices.

3. EXPLOITING THE DEVICE-TO-DEVICE COMMUNICATIONS

In traditional networks, mobile devices are forced to communicate through base stations, and they are not allowed to share data directly between them. With the introduction of a myriad of smart handheld devices (in particular smartphones), user demands for mobile broadband are undergoing an unprecedented rise. The drastic growth of bandwidth-hungry applications such as video streaming and multimedia file sharing are already pushing the limits of current cellular systems [20].

Mobile architectures, which exploit cooperative paradigm, represent a new class of wireless communication techniques in which devices help each other in doing complex and heavy tasks, such as extract high level information from raw signals. The term device here refers to a cell phone or any other portable wireless device with cellular connectivity (tablet, laptop, etc) a user owns. This new transmission paradigm promises significant performance gains in terms of spectral efficiency, latency, transmission range and information sharing.

One possible way to realise this new paradigm is through the use of device-to-device (D2D) communications functionality, which allows two nearby devices to communicate with each other in the licenced cellular bandwidth without a base station (BS) involved or with limited BS involvement [21, 22]. In former network generations, D2D has not been considered. Consequently, with the approach of 5G networks, the possibilities of exploiting such technology are quite large.

3.1. WiFi direct

An important and practical example of D2D technology is represented by WiFi Direct, also called WiFi Peer-to-Peer (P2P). It is a wireless standard enabling mobile devices to connect with each other without requiring a BS or other technologies (such as Bluetooth). This particular technology does not have a specific final application but it is exploitable for everything, from Internet browsing to data and information transfer. Specifically, it allows for communicating with more than one device simultaneously, enabling connections between devices even if they are from different manufacturers. Indeed, exploiting direct communication between nearby mobile devices will improve spectrum utilisation, overall throughput and energy efficiency, all while enabling context-aware-based applications and services.

Such new technology has several gains and technical solutions. Some of them include, but are not limited to [23]:

- Capacity gain: because of the possibility of sharing spectrum resources between cellular and D2D users.
- User data rate gain: because of the close proximity and potentially favourable propagation conditions high peak rates may be achieved.
- Latency gain: when devices communicate over a direct link, the end-to-end latency may be reduced.

From a practical viewpoint, WiFi Direct devices (formally known as *P2P Devices*) communicate by establishing *P2P Groups*, which are functionally equivalent to traditional WiFi infrastructure networks [24]. The communication bandwidth can reach a maximum peak of 250 [MBps] [25].

As detailed in the next section, this paper proposes one important and practical application of the D2D communication called TCAC. We propose a new architecture that aims to exploit the D2D communication by using a WiFi Direct approach, enabling devices present in an environment to directly share high-level information (e.g. the gender and/or the identity of a speaker) rather than simply their physical resources (e.g. CPU, memory and network).

Every smartphone, laptop or tablet can overcome its own physical limitations through utilising the capabilities offered by nearby devices over an ad hoc network. Coupled with the fact that 5G networks will provide all the users the possibility to exploit the D2D communications, this specific architecture is well-suited as an enabling technology for the next generation of mobile networks.

4. THE TRANSIENT CONTEXT-AWARE CLOUD

4.1. Transient clouds

Transient clouds capitalise on the fact that smart devices can be more useful and more effective when they work together. Because of the ubiquitous nature of these devices and their great diffusion, it is reasonable to think that wherever there is a crowd of people there will also be a large number of devices present. Importantly, these devices are not homogeneous, but each has its own unique features. TCs are designed to exploit this local non-conformity considering that the devices in the area connect to each other over an ad hoc network while offering different capabilities for others to use. This network functions in a way similar to peer-to-peer networks, without relying on an infrastructure.

The capability that a device offers can be generic. Typically, devices can advertise their hardware features, such as having a camera or a GPS chip, but software defined features such as specific algorithms that can be executed can be capabilities as well. A device in a TC can simply request a specific capability, and a device in the TC offering that capability will be recruited to help out. As an example, consider a device without a GPS chip that would like to learn its location. It can simply ask the TC, and a nearby device

offering a GPS chip as a capability can provide its location as an approximate location to the requesting device.

By offering these capabilities, not only can more tasks be done by more devices but also the workload can be shared across all the devices in the TC by offloading sections of code to other devices. This sharing can prolong the life of the devices in the network, rather than relying so heavily on a small number of devices that will die quickly from being overworked. For more details, we refer the reader to [4].

4.2. Context-awareness as smartphone capability

The open, programmable and ever-growing sensor on mobile devices have enabled new sensing applications to be created across a wide variety of domains such as social networks, mobile health, gaming, entertainment, education and transportation. Devices present three different capabilities: communication capability, sensing capability and information processing capability.

As reported in [26], they can be described by the three-trunk "hub+sensor+processor" paradigm. Specifically, the *hub* trunk presents both short-range (Bluetooth and WiFi employed for local information exchange) and long-range (GPRS, 3G/4G and WiFi employed as Internet access) communication capability. The *sensor* trunk is implemented through sensors embedded into smartphones such as GPS receivers, accelerometers, microphones and

radio interfaces. Finally, the *information processing* trunk is represented by the smartphones' CPUs, which support flexible and powerful software. Smartphones can infer context information by acquiring data from the low level embedded sensors (microphone or accelerometer), process it to extract high-level information (who is speaking, rather than which kind of movement the user is performing), and send it to a final destination over a telecommunication network. By exploiting all these capabilities, smartphones can become powerful augmented sensors able to sense, process, and transmit high-level information, and able to characterize the situation of an entity or a group of entities and to provide information about the present status of the entities. The involved entities can be people, machine devices, objects or locations. This new smartphone extended capability is also known as context-awareness [3].

4.3. Transient context-aware cloud

The basic idea of TC relies on the concept of collaborative computing that allows nearby devices to form an ad hoc network to provide various basic capabilities as a cloud service [4]. As reported in Figure 1, TC permits dividing low-level tasks, leaving the network to distribute the computational load of extracting high-level information. On one hand, this could allow the users to save smartphones' battery lifetime and processing power. On the other hand,

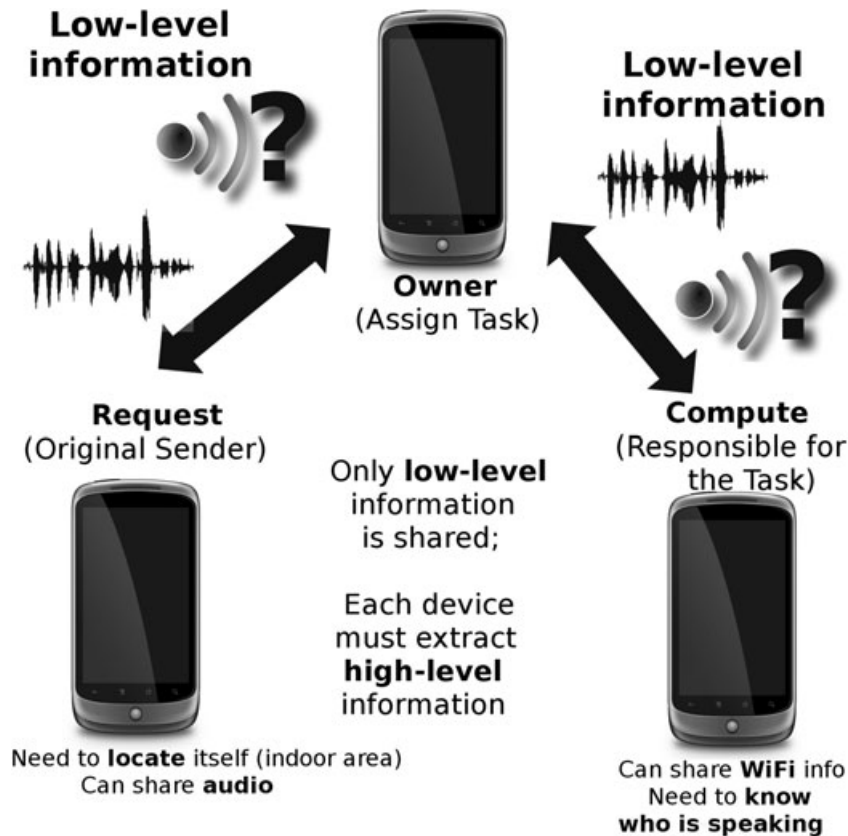


Figure 1. Structure of the traditional transient cloud architecture.

it has the limitation of only considering low-level tasks (such as sharing acquired audio or WiFi information). In this work, we merge context-aware capabilities offered by smartphones with the TC paradigm. Specifically, we have employed a solution able to recognise the speaker's main characteristics, similarly to what already done in [5]. It uses a supervised classifier jointly with some specific audio features. Together with this audio process application, we have employed the TC architecture, described in [4].

We call this new paradigm TCAC. Each mobile device involved in the TCAC acquires the raw data from the embedded sensors and infers high-level information, which can be directly shared with other smartphones inside the TCAC platform, as reported in Figure 2. This idea has several important and practical applications. Considering, for example, the meeting scenario in Section 1, in which access to the network is forbidden (or discouraged) for privacy issues or it is simply unavailable: the TCAC platform could exploit high-level information extracted and shared by other mobile devices. If several devices are in the same room and one of them recognises who is speaking, it can send this information through the TCAC platform, so that all the other devices could access it. Furthermore, if one smartphone infers which room the meeting is in by using an indoor positioning algorithm, all

the other devices will know their position without having to compute it. This allows the devices to save resources such as energy, memory and battery life.

5. AN ANALYTICAL MODEL OF ENERGY CONSUMPTION

One of the key points of the proposed architecture is to exploit the TCAC to efficiently share high-level information among the devices within the cloud. For this reason, we propose an analytical model, which takes into account the energy required by a single smartphone to process the amount of operation requested by the TCAC.

We model the energy necessary (expressed in [J]) for a single device within the cloud as the sum of three contributions: (i) the energy necessary to receive the data from the cloud E_{RX} ; (ii) the energy necessary to process the data E_P ; and (iii) the energy needed to send the data back within the cloud E_{SN} .

$$E_{TCAC} = E_{RX} + E_P + E_{SN} \quad (1)$$

Each components of Equation (1) can be further decomposed as follows:

$$E_{RX} = P_{RX} \cdot T_{RX} \quad (2)$$

$$E_P = P_P \cdot T_P \quad (3)$$

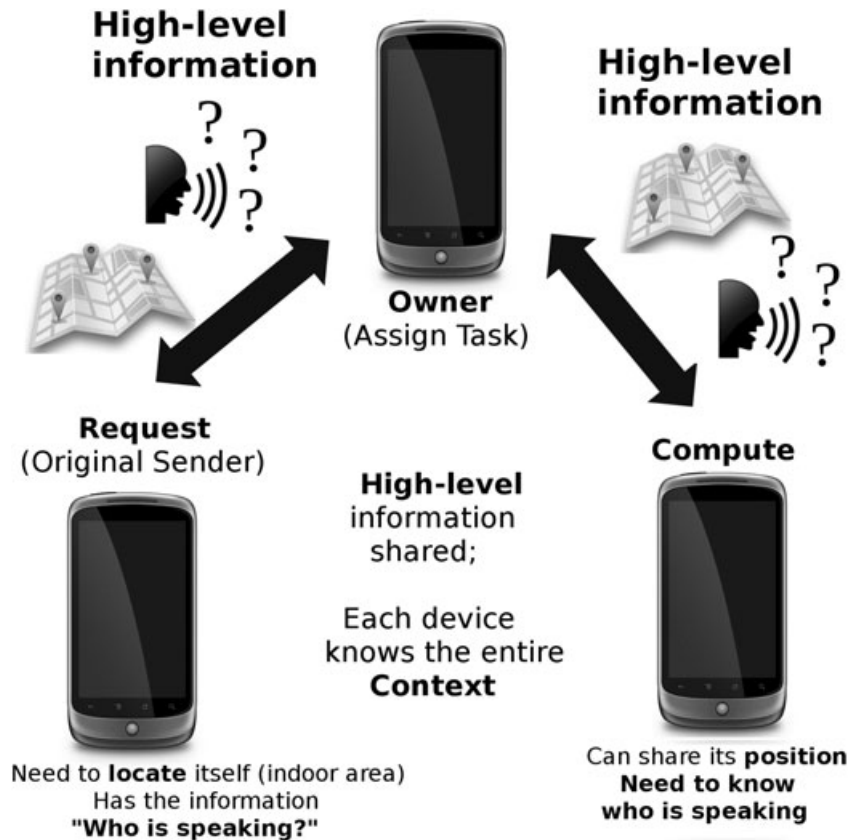


Figure 2. Structure of the transient cloud-based context-awareness architecture.

$$E_{SN} = P_{SN} \cdot T_{SN} \quad (4)$$

where P_{RX} , P_P and P_{SN} are the power (expressed in [W]) necessary to perform the reception, the processing and to send data back to the cloud, respectively. Similarly, T_{RX} , T_P and T_{SN} are the times (in [s]) that every single contribution requires in order to be completed.

The times necessary to receive and to send back data from and to the cloud can be further decomposed.

$$E_{RX} = P_{RX} \cdot T_{RX} = P_{RX} \cdot \frac{D_{RX}}{B_{RX}} \quad (5)$$

$$E_{SN} = P_{SN} \cdot T_{SN} = P_{SN} \cdot \frac{D_{SN}}{B_{SN}} \quad (6)$$

where D_{RX} and D_{SN} are the total amount of data (in [KB]) received from and sent to the cloud, respectively. The quantity B_{RX} is the transmission bandwidth while B_{SN} is the receiver bandwidth. Both of them are expressed in [KBps].

Consequently, the overall energy consumed by a single mobile device within the TCAC can be written as follows:

$$E_{TCAC} = P_{RX} \cdot \frac{D_{RX}}{B_{RX}} + P_P \cdot T_P + P_{SN} \cdot \frac{D_{SN}}{B_{SN}} \quad (7)$$

From Equation (7) it can be seen that the quantity E_{TCAC} is composed of three different factors. The first and the third ones are the energy necessary to receive and to send back data to and from the TCAC. These form the Communication Energy. The second one is the energy required by the mobile device to process the data, referred to as Computation Energy.

In the next subsections, each of these quantities will be deeply analysed and detailed.

5.1. The computation energy

In this subsection, the computation energy necessary to process the data by the smartphone within the cloud is considered. Thanks to the Android Power Tutor tool, we can consider the power P_P necessary to activate the CPU to be known (see Section 6 for details). As a consequence, all the information necessary is related to the time T_P . Obviously, this quantity is a function of D_{RX} . Indeed, the more data are transmitted and processed, the longer the time necessary. Furthermore, T_P also depends on the characteristics of the mobile device (more powerful smartphones require less time) and on the type of operations that must be executed (heavier calculations demand more computation time). In more detail:

$$T_P \longrightarrow F(D_{RX}, \Theta, \Psi) \quad (8)$$

The quantity Θ represents the vector, which models the single smartphone's characteristics. It takes into account the important parameters such as CPU power, memory, battery charge and so on. The other quantity Ψ is the task's parameter vector. In practice, it weights the heaviness of the operations that the task requires.

Table I. Different values of the time T_P .

Samples per frames	Numbers of frames	D_{RX} [Kbyte]	T_P [ms]
320	442	1132	245
320	367	940	249
318	172	438	203
319	183	467	139
320	76	195	95
322	72	185	75
318	81	206	153
319	354	903	223
320	177	453	160
320	264	676	172

The function $F : \mathbb{R}^{\|\Theta\| + \|\Psi\| + 1} \mapsto \mathbb{R}$ represents a mathematical relationship which takes as inputs the smartphone parameters vector and the amount of data received by the smartphone itself and outputs the time necessary to process it when a smartphone which has such characteristics is employed.[†]

The formal definition and the estimation of these two vectors are out of the scope of this paper. Nevertheless, in order to obtain valid results for the time T_P , we always refer to the same smartphone (so that Θ is not changing) and to same operations (so that Ψ is constant). From a mathematical viewpoint, this leads to

$$T_P \longrightarrow F(D_{RX}) \quad (9)$$

In this paper, some empirical values of T_P have been measured by using an off-the-shelf smartphone and reported in Table I. From the aforementioned table, it is possible to see a direct correlation between the size (expressed in [Kbyte]) of the data and the time needed to process it. To obtain an exploitable mathematical relationship between these two quantities, a least-square (LS) third-degree polynomial interpolation has been employed. It is reported in the following formula:

$$T_P = 1.3 \cdot 10^{-6} \cdot D_{RX}^3 - 3.24 \cdot 10^{-4} \cdot D_{RX}^2 + 0.38 \cdot D_{RX} + 44.21 \quad (10)$$

Figure 3 reports the values of the time T_P for different size of D_{RX} (single points) and their polynomial interpolation (continuous blue line).

The formula reported in Equation (10) will be exploited in the next sections to show the benefits, in terms of amount of energy saved, of employing the TCAC with respect to using only one single mobile device.

5.2. The communications energy

This subsection considers the energies, which compose the communication energy. The first one is from the transmission of data from the TCAC to the single device,

[†]The function $\|\mathbf{V}\|$ returns the length of the vector \mathbf{V} .

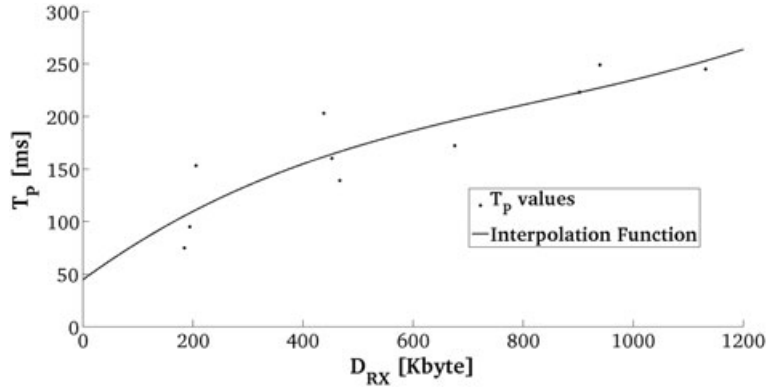


Figure 3. Values of the time T_P for different size of D_{RX} (single points) and a third degree interpolation (continuous blue line).

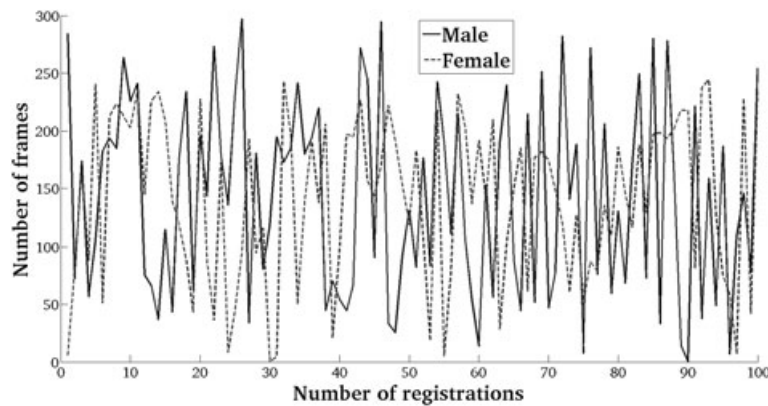


Figure 4. Number of audio frames employed during the audio processing task for a male speaker (continuous blue line) and a female one (dashed red line).

while the second one is from the opposite operation: the device sending the data back to the TCAC. As reported in Equations (5) and (6), they are functions of the amount of data received D_{RX} and sent back D_{SN} . The data received to be processed can be estimated by considering Figure 4, which shows that the number of frames employed in the audio processing task is in the range of [50 – 300].

The proposed gender recognition audio task employs 26 features. Each of them is represented and stored by using a 64 bit double-precision format. As a consequence, the data received can be estimated within a interval, which roughly ranges from $\frac{26 \cdot 64 \cdot 50}{8 \cdot 1000} \approx 10$ [Kbyte] to $\frac{26 \cdot 64 \cdot 300}{8 \cdot 1000} \approx 65$ [Kbyte]. Considering the amount of data sent back to the TCAC, the considered values are significantly smaller. Specifically, they range in the interval between [1 – 5] Kbyte.

The plot reported in the Figure 5 shows the trend of the communication energy $E_{Comm} = E_{RX} + E_{SN}$ (expressed in [J]) when different values of D_{RX} and D_{SN} are considered.

From the plot, important observations can be made. The first one concerns the fact that the energy is a function, which grows with the increasing of D_{RX} and D_{SN} . Furthermore, D_{RX} is a quantity more significant with respect to D_{SN} . If a specific value of D_{RX} is fixed and D_{SN} varies

(i.e. pick up a single line of the plot), there is a very small variation in terms of the energy requested. On the other hand, if a specific value of D_{SN} is fixed and D_{RX} varies, the variation of the energy E_{Comm} is more significant. This is motivated by the fact that the amount of data D_{RX} processed by a device within the TCAC is greater than the amount of data D_{SN} that, from the same device, will sent back to the TCAC when the process is over.

6. PERFORMANCE EVALUATION

6.1. The test-bed setup

To test the validity of our approach, we have employed smartphones from four different mobile device classes for the proposed experiments. Their main technical specifications are reported in Table II. The smartphones are connected together using WiFi Direct to create a TC that allows nearby devices to share high-level information. All the experiments were conducted by connecting the devices to the group owner. When the TCAC platform has extracted the speaker's details, the information is propagated throughout the cloud, so that every other smartphone can exploit it.

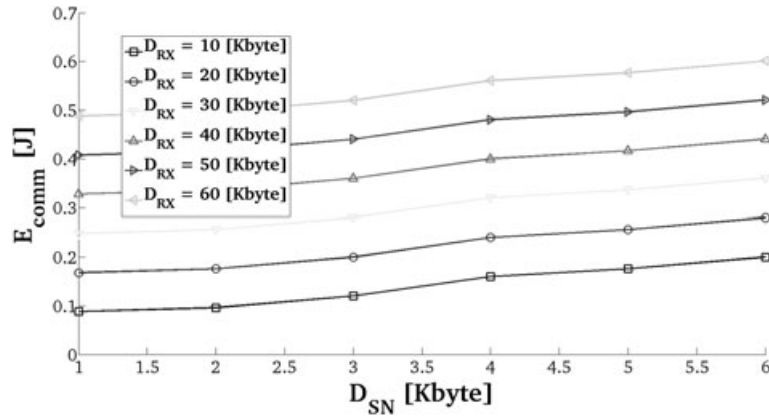


Figure 5. Trends of the theoretical energy E_{Comm} for a single device employed within the transient context-aware cloud.

Table II. Technical specifications of the smartphones used.

Specs	Mobile device classes			
	Class 1	Class 2	Class 3	Class 4
CPU	Snapdragon 600	Snapdragon 800	Exynos 4412	Exynos 4612
	1.9 GHz	2.3 GHz	1.4 GHz	1.6 GHz
Memory	32 GB	32 GB	16 GB	32 GB
	2 GB RAM	2 GB RAM	1 GB RAM	1 GB RAM
Android	4.4.2 (KitKat)	4.4.4 (KitKat)	4.3 (JellyBean)	4.1 (JellyBean)
	9.98	9.88	7.98	11.78
Total charge [Wh]				

6.2. Power and energy metrics

In this subsection, the difference in terms of the amount of power required by the local computation and the TCAC approach is evaluated and discussed. In more detail, the power, expressed in milliwatt [mW], necessary for a single device to acquire audio samples from the smartphone’s

microphone and extract the features has been measured. The results are shown in Figure 6.

To measure the power consumed by a smartphone, we have written an ad hoc Java programme, which links with the Android debug bridge. The Android debug bridge is a versatile command line tool that allows users communicating with a connected Android-powered device. It is a client-server programme that includes three components: (i) a client (which runs on your development machine); (ii) a server (which runs as a background process on your development machine); and (iii) a daemon (which runs as a background process on the device) [27].

Figure 6 shows the details in terms of power consumed when all the calculations are performed locally. In this plot, the measurements related to certain operations are highlighted. The average value of the whole operation is about 1150[mW] (dashed red line). It is worth remembering that this value is related to the power necessary for one smartphone to infer a speaker’s high-level information. If more mobile devices with similar characteristics are considered, it is a reasonable assumption that each of them will employ a similar amount of power. Consequently, the

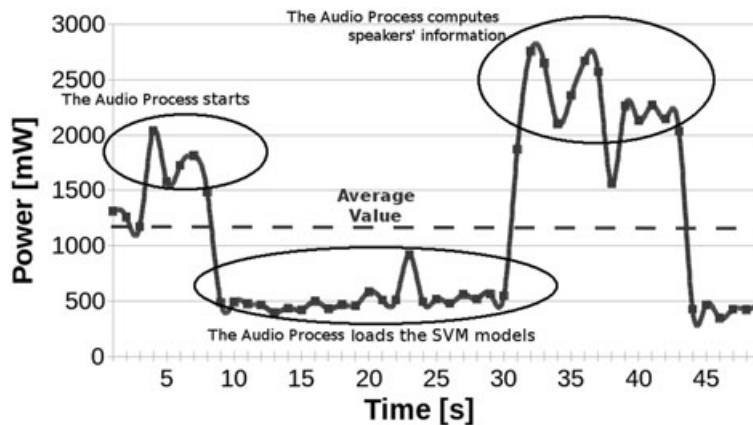


Figure 6. Power trend of a single smartphone, which computes the calculations demanded by the context-aware audio process locally (continuous blue line). The average value is about 1150[mW] (dashed red line).

average power requested by each smartphone to extract the speaker's information should be multiplied by the number of smartphones involved in the experiments in order to obtain the total power necessary for every device to have the same high-level information. This fact translates into an inefficient exploitation of the available devices' energy.

When a TCAC approach is employed, high-level information (i.e. the gender of the speaker, in the case of this paper) is automatically propagated to all the devices involved in the cloud. The result reported in Figure 7 shows the advantage of the proposed approach with respect to the case in which all the mobile devices compute their calculations locally. In particular, Figure 7 reports the energy in [mWh] necessary to extract the speaker's information when (i) all the calculation is performed locally on a single smartphone (red bins) and (ii) the calculations are shared within the TCAC (blue bins), as a function of the number of smartphones involved in the experiment. From the bars reported in the plot it is immediately clear that, when three or more smartphones are employed, the TCAC allows for significant savings of energy, extending the devices' battery lifetime. Specifically, directly from Figure 7, it is possible to infer that the amounts of energy saved are around 20.6%, 26.47% and 55.90% when three, four and five smartphones are employed, respectively.

The same result can be also achieved by employing the analytical model proposed in Section 5. In order to do so, the amount of energy consumed when a single device is employed (Equation (7)) must be compared with the amount of energy consumed when more devices are considered. For the sake of simplicity, we assume that, if more than one device is considered, the overall amount of data (D_{RX} and D_{SN}) is equally divided among all the devices involved in the TCAC.

Let N be the number of smartphones within the proposed cloud. The employment of the TCAC with respect to the single-device approach is convenient, in terms of energy saved, if the inequality in Equation (11) holds in which we have substituted the value of T_P from Equation (10) within Equation (7), and we have divided the overall amount of

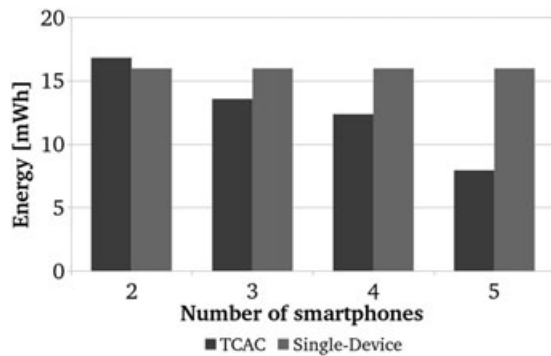


Figure 7. Comparison of the energy consumed [mWh] as a function of the number of smartphones when (i) all the calculation are performed locally (red bins), and (ii) all the calculations are shared within the transient context-aware cloud (blue bins).

data (D_{RX} and D_{SN}) by the number of devices N considered within the TCAC. Specifically, the left part of Equation (11) is the total energy consumed when just one smartphone is considered, while the right part is the energy necessary when N devices are employed. Moving all the contribution to the left part and doing some calculations provides the result reported in Equation (12). The advantage, in terms of saved energy when more than one smartphone is employed, will translate in having positive values of Equation (12).

Considering that $N > 0$, we can multiply both left and right parts of Equation (12) by N^3 and, after some algebraic manipulations, we obtain Equation (13), which can be solved numerically. To do so, we have employed the following values: $P_{RX} = P_{SN} = 0.5[mW]$, $B_{RX} = B_{SN} = 250[MBp]$, $D_{RX} = 5[Kbyte]$, $D_{SN} = 1[Kbyte]$ and $P_P = 0.1[mW]$. Table III shows some values of Equation (13) for different numbers of smartphones N employed within the TCAC. Accordingly with what already shown and commented on in Figure 7, using three or more smartphones allows the TCAC to save energy and, as a consequence, extend the device battery lifetime.

$$\begin{aligned}
 & P_{RX} \frac{D_{RX}}{B_{RX}} + P_P \left(1.3 \cdot 10^{-6} D_{RX}^3 - 3.24 \cdot 10^{-4} D_{RX}^2 \right. \\
 & \quad \left. + 0.38 D_{RX} + 44.21 \right) + P_{SN} \frac{D_{SN}}{B_{SN}} > \\
 & > P_{RX} \frac{D_{RX}}{N \cdot B_{RX}} + P_P \left(1.3 \cdot 10^{-6} \frac{D_{RX}^3}{N} - 3.24 \cdot 10^{-4} \frac{D_{RX}^2}{N} \right. \\
 & \quad \left. + 0.38 \frac{D_{RX}}{N} + 44.21 \right) + P_{SN} \frac{D_{SN}}{N \cdot B_{SN}} \quad (11)
 \end{aligned}$$

$$\begin{aligned}
 & \frac{N-1}{N} \left(P_{RX} \cdot \frac{D_{RX}}{B_{RX}} \right) + \frac{N-1}{N} \cdot P_{SN} \cdot \frac{D_{SN}}{B_{SN}} + P_P \cdot D_{RX} \\
 & \times \left(1.3 \cdot 10^{-6} \frac{N^3-1}{N^3} - 3.24 \cdot 10^{-4} \frac{N^2-1}{N^2} + 0.38 \frac{N-1}{N} \right) > 0 \quad (12)
 \end{aligned}$$

Table III. Some numerical values of Equation (13). When $N \geq 3$ the values become positive, so yielding an improvement in terms of saved energy when just one smartphone is employed.

N	Equation (13)
1	-1.31
2	-0.002
3	3.56
4	10.49
5	21.91

$$\begin{aligned}
 & N^2(N-1) \left[P_{RX} \cdot \frac{D_{RX}}{B_{RX}} \right] \\
 & + P_P \cdot D_{RX} \left[1.3 \cdot 10^{-6}(N^3 - 1) - 3.24 \cdot 10^{-4}N(N^2 - 1) \right. \\
 & \left. + 0.38N^2(N-1) \right] > 0
 \end{aligned} \tag{13}$$

The next shown result deals with the power consumed by a single smartphone when it is involved in the TCAC. Reprising the TC architecture, reported in both Figures 1 and 2, we consider the power consumed by a mobile device when it covers different roles within the TCAC. Figure 8 shows the power consumption when the smartphone is the TCAC *Owner* device (average value 500[mW]), Figure 9 presents the power consumption for a mobile device which demands (*Request* device) the context-aware task (average value 675[mW]) and, finally, Figure 10 reports the power consumption when the smartphone is chosen to compute the final high-level information (*Compute* device, average value 101[mW]). The overall average power of three components (*Owner*, *Request* and *Compute*) of the TCAC is $\frac{500+675+101}{3}$ [mW] = 425[mW].

The *Owner* device (Figure 8) presents peaks within the first seconds, due to the WiFi Direct connection management. The *Request* device and the *Compute* device

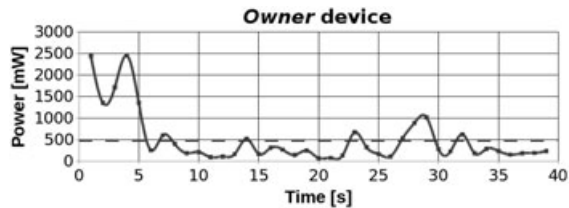


Figure 8. Power consumption [mW] of the Owner device. The average value is around 500 [mW] (red dashed line).

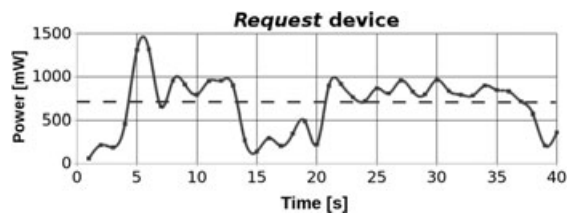


Figure 9. Power consumption [mW] of the Request device. The average value is around 675 [mW] (red dashed line).

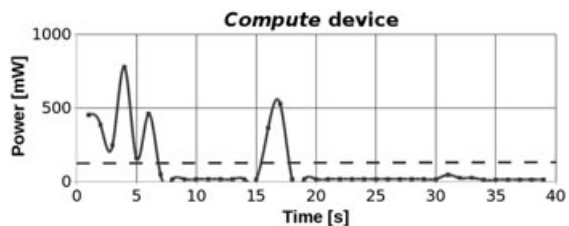


Figure 10. Power consumption [mW] of the Compute device. The average value is around 101[mW] (red dashed line).

have opposite power trends. From Figures 9 and 10, it is worth noting that within the interval between 15 and 20s there is a drop in the power consumption of the *Request* device (Figure 9) and, in the same temporal interval, the *Compute* device exhibits a peak in its power consumption (Figure 10). This can be easily explained by considering that when the *Compute* device contributes to infer the high-level information, the *Request* device saves power by offloading some computation. The three plots reported previously further confirm that when three or more smartphones collaborate within a TCAC they are able to significantly save the power consumed. In fact, when the requested context-aware task is finished, each device will have the available final high-level information, with a power consumption for each device significantly lower than 1150[mW] necessary for a single smartphone, which computes the calculations locally (Figure 6).

6.3. Accuracy of the context-aware algorithm

When the TC meets a context-aware algorithm, the accuracy, in terms of the capacity of the employed solution to perform correctly, must be carefully addressed. Here, the authors have employed an algorithm able to infer some of the main speaker's characteristics (e.g. gender, identity and language) from his voice signal (inspired by [5] for all the technical details). For the sake of brevity, this work only considers the gender recognition capability, but results and conclusions can be easily extended to the other cases of identity and language. Obviously, considering that the TCAC only shares high-level information, employing the gender recognition algorithm on TCAC platform does not affect in any way the performance in terms of accuracy of the aforementioned algorithm with respect to its stand-alone usage. In other words, the results showed does not depend in any case on the number of the smartphones involved within the TCAC.

Table IV reports the accuracy percentage of the gender recognition algorithm embedded within the TCAC when different audio features are employed for the cross validation and the open set scenario, respectively. Specifically, we have used Mel frequency cepstrum coefficients , linear

Table IV. Accuracy percentage for context-aware task (gender recognition) performed by audio process task.

	Accuracy(%)	
	Cross validation	Open set
LPC	93.89	92.8
LPC + Δ	95.37	91.86
LPC + ΔΔ	97.43	94.01
MFCC	98.31	96.78
MFCC + Δ	98.76	97.23
MFCC + ΔΔ	98.9	97.45
MFCC + LPC	96.28	93.45
MFCC + LPC + Δ	98.9	97.45
MFCC + LPC + ΔΔ	99.56	98.25

predictive coding coefficients, delta coefficients (Δ) and Delta-Delta coefficients ($\Delta\Delta$).

From Table IV, it is worth noting that the gender recognition algorithm shows good performances (around 98% for the open set scenario) for all the employed features. Furthermore, the table also highlights that the more features are employed, the better the accuracy percentage.

7. CONCLUSIONS

In many scenarios, it is important for the nodes in the network to compute and agree on a particular context. In some cases, it would not be worthwhile for multiple nodes to go through the same process to compute a context and in other cases, it may not be feasible. In this paper, we have presented an architecture in which context awareness can be provided over a TC. We called this new paradigm TCAC. This idea exploits the D2D communications, which allows two nearby devices to communicate with each other in the licenced cellular bandwidth without a BS involved.

It is worth noticing that the proposed paradigm can be implemented by using a WiFi Direct approach but it can be portable through any network, which exploits the D2D communications, so opening the doors to all the forthcoming 5G scenarios.

We propose analytical, numerical and measurement based results, obtained by physically measuring the power consumed by the employed smartphones. In addition, also an analytical model of the energy necessary to offload some computation to another device within the cloud was provided and detailed.

Our results, both the theoretical model and practical measures, show that when a few smartphones (e.g. three in our evaluation) are involved in the TCAC, there is a significant saving in terms of energy required to perform the same calculations, allowing for a significant extension of the devices battery lifetimes.

Furthermore, practical experiments show that a single device, which computes the computation locally requires about 1150[mW] while, when the same smartphone is employed within the TCAC, it needs an average value around 425[mW].

REFERENCES

- Gordon M, Jamshidi D, Mahlke S, Mao Z, Chen X. COMET: Code Offload by Migrating Execution Transparently. In *Proceedings of OSDI*, Hollywood, CA, 2012; 93–106.
- Kemp R, Palmer N, Kielmann T, Bal H. Cuckoo: A Computation Offloading Framework for Smartphones. In *Mobile Computing, Applications, and Services*, Vol. 76, Gris M, Yang G (eds), Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer: Berlin Heidelberg, 2012; 59–79.
- Wu J, Bisio I, Gniady C, Hossain E, Valla M, Li H. Context-aware networking and communications: part 1 [guest editorial]. *IEEE Communications Magazine* 2014; **52**(6): 14–15.
- Penner T, Johnson A, Van Slyke B, Guirguis M, Gu Q. Transient clouds: Assignment and collaborative execution of tasks on mobile devices. In *2014 IEEE Global Communications Conference*, Austin, TX, 2014; 2801–2806.
- Bisio I, Lavagetto F, Marchese M, Sciarrone A, Frá C, Valla M. SPECTRA: A SPEech proCessing plaTform as smaRtphone Application. In *IEEE International Conference on Communications (ICC'15)*, London, UK.
- Abowd GD, Dey AK, Brown PJ, Davies N, Smith M, Steggle P. Towards a better understanding of context and context-awareness. In *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing*, ser. HUC '99. Springer-Verlag: London, UK, UK, 1999; 304–307.
- Bisio I, Lavagetto F, Marchese M, Sciarrone A. Smartphone-based user activity recognition method for health remote monitoring applications. In *PECCS*, Benavente-Peces C, Ali FH, Filipe J (eds). SciTePress: Rome, Italy, 2012; 200–205.
- Yurur O, Liu C, Sheng Z, Leung V, Moreno W, Leung K. Context-awareness for mobile sensing: a survey and future directions. *IEEE Communications Surveys Tutorials* 2014; **99**: 1–1.
- Bisio I, Delfino A, Lavagetto F, Marchese M, Sciarrone A. A smartphone-centric platform for remote health monitoring of heart failure. *International Journal of Communication Systems* 2015; **28**(11): 1753–1771.
- Bisio I, Lavagetto F, Marchese M, Sciarrone A. Gps/hps and wi-fi fingerprint-based location recognition for check-in applications over smartphones in cloud-based lbs. *IEEE Transactions on Multimedia* 2013; **15**(4): 858–869.
- Bisio I, Delfino A, Lavagetto F, Marchese M, Sciarrone A. Gender-driven emotion recognition through speech signals for ambient intelligence applications. *IEEE Transactions on Emerging Topics in Computing* 2013; **1**(2): 244–257.
- Cuervo E, Balasubramanian A, Cho D-k, Wolman A, Saroiu S, Chandra R, Bahl P. Maui: Making smartphones last longer with code offload. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '10. ACM: York, NY, USA, 2010; 49–62.
- Miluzzo E, Cáceres R, Chen Y-F. Vision: Mclouds - computing on clouds of mobile devices. In *Proceedings of the Third ACM Workshop on Mobile Cloud Computing and Services*, ser. MCS '12. ACM: New York, NY, USA, 2012; 9–14.

14. Langford T, Gu Q, Rivera-Longoria A, Guirguis M. Collaborative computing on-demand: Harnessing mobile devices in executing on-the-fly jobs. In *2013 IEEE 10th International Conference on Mobile Ad-Hoc and Sensor Systems (MASS)*, Hangzhou, China, 2013; 342–350.
15. Guirguis M, Ogden R, Song Z, Thapa S, Gu Q. Can you help me run these code segments on your mobile device? In *2011 IEEE Global Telecommunications Conference*, Houston, TX, 2011; 1–5.
16. Tsai C-W, Rodrigues J. Metaheuristic scheduling for cloud: a survey. *IEEE Systems Journal* 2014; **8**(1): 279–291.
17. Liu J, Kawamoto Y, Nishiyama H, Kato N, Kadowaki N. Device-to-device communications achieve efficient load balancing in lte-advanced networks. *IEEE Wireless Communications* 2014; **21**(2): 57–65.
18. Kumar N, Iqbal R, Misra S, Rodrigues JJ. Bayesian coalition game for contention-aware reliable data forwarding in vehicular mobile cloud. *Future Generation Computer Systems* 2015; **48**(C): 60–72.
19. Nishiyama H, Ito M, Kato N. Relay-by-smartphone: realizing multihop device-to-device communications. *IEEE Communications Magazine* 2014; **52**(4): 56–65.
20. Mohsen Nader Tehrani MU, Yanikomeroglu H. Device-to-device communication in 5g cellular networks: Challenges, solutions, and future directions. *IEEE Communications Magazine* 2014; **52**(5): 86–92.
21. Militano L, Condoluci M, Araniti G, Molinaro A, Iera A, Fitzek F. Wi-fi cooperation or d2d-based multicast content distribution in lte-a: A comparative analysis. In *2014 IEEE International Conference on Communications Workshops (ICC)*, Sydney, NSW, 2014; 296–301.
22. Militano L, Condoluci M, Araniti G, Molinaro A, Iera A. When d2d communication improves group oriented services in beyond 4g networks. *Wireless Networks* 2015; **21**(4): 1363–1377.
23. Mumtaz S, Rodriguez J. Introduction to D2D communication. In *Smart Device to Smart Device Communication*. Springer International Publishing: Switzerland, 2014.
24. Camps-Mur D, Garcia-Saavedra A, Serrano P. Device-to-device communications with wi-fi direct: overview and experimentation. *IEEE Wireless Communications* 2013; **20**(3): 96–104.
25. Feng D, Lu L, Yuan-Wu Y, Li G, Li S, Feng G. Device-to-device communications in cellular networks. *IEEE Communications Magazine* 2014; **52**(4): 49–55.
26. Bisio I, Lavagetto F, Marchese M, Sciarrone A. Smartphone-centric ambient assisted living platform for patients suffering from co-morbidities monitoring. *IEEE Communications Magazine* 2015; **53**(1): 34–41.
27. Android debug bridge. Available from: <http://developer.android.com/tools/help/adb.html> [25 May 2015].