

# Exploiting LTE D2D Communications in M2M Fog Platforms: Deployment and Practical Issues

Carlo Vallati, Antonio Viridis, Enzo Mingozzi, Giovanni Stea  
Dipartimento di Ingegneria dell'Informazione, University of Pisa  
Largo Lucio Lazzarino 1, I-56122, Pisa, Italy  
{c.vallati, a.viridis, e.mingozzi, g.stea}@iet.unipi.it

**Abstract**—Fog computing is envisaged as the evolution of the current centralized cloud to support the forthcoming Internet of Things revolution. Its distributed architecture aims at providing location awareness and low-latency interactions to Machine-to-Machine (M2M) applications. In this context, the LTE-Advanced technology and its evolutions are expected to play a major role as a communication infrastructure that guarantees low deployment costs, plug-and-play seamless configuration and embedded security. In this paper, we show how the LTE network can be configured to support future M2M Fog computing platforms. In particular it is shown how a network deployment that exploits Device-to-Device (D2D) communications, currently under definition within 3GPP, can be employed to support efficient communication between Fog nodes and smart objects, enabling low-latency interactions and locality-preserving multicast transmissions. The proposed deployment is presented highlighting the issues that its practical implementation raises. The advantages of the proposed approach against other alternatives are shown by means of simulation.

**Keywords**—M2M; Fog Computing; LTE; LTE-Advanced; D2D;

## I. INTRODUCTION

Cloud computing represents nowadays the most popular source of low-cost and easy-deployable computational power and storage. Although the cloud infrastructure supports a wide range of applications, its fully centralized paradigm is expected to be inefficient considering the future Internet of Things (IoT). The current architecture that processes all the data in the cloud and runs applications far from physical deployments prevents the development of Machine-to-Machine (M2M) applications that require *low-latency* and *locality-preserving* communication among physical objects. To this aim, a novel distributed architecture is envisaged to enable local intelligence to be deployed close to IoT systems. Such evolution, referred to as *Fog computing* [1], leverages the deployment of *Fog Nodes* installed directly on the field among sensors and actuators to enable the execution of simple M2M applications. Direct interaction between the Fog Node and the things enables the deployment of M2M applications that have stringent real-time QoS requirements, e.g. real-time closed-loop control applications [2], or that require true local interactions, e.g. applications based on opportunistic local interactions [3]. As an example, let us consider a smart-home platform for home security and energy efficiency. The Fog node running the intrusion detection application will require timed and reliable communication with intrusion detection sensors and door locks, while the management application that leverages opportunistic interactions to coordinate appliances will require automated discovery of new devices. In this context, 4G/5G cellular networks, and specifically LTE-Advanced and its evolutions, will play a major role, due to its widespread, ubiquitous coverage [4]. Although different communication technologies for IoT are available, e.g. IEEE 802.15.4 or Bluetooth low-power, LTE is expected to be a major player considering the widespread infrastructures already deployed in a large scale and the service reliability offered by a mature wireless standard deployed on licensed spectrum [12]. In addition, LTE offers by design several features mandatory to support M2M communications, such as auto-configuration for plug-and-play devices that do not require human intervention, security through data encryption and authentication, Quality of Service (QoS) support for data delivery with stringent real-time requirements, and energy-saving mechanisms for battery-powered devices [17]. Besides the classical Device-to-Infrastructure (D2I) communication, the LTE standard includes a Device-to-Device (D2D) mode that supports direct interaction between devices without relaying the communication through the base-station node. Such mode of operation has been recently introduced to further support M2M applications, facilitating the discovery of neighboring devices and direct communication with reduced costs. Although the LTE standard offers several technical solutions amenable to M2M interactions, it does not suggest one practical solutions specifically [20].

In this paper, we show how LTE D2D communications can be exploited in M2M platforms based on Fog computing. Considering a generic M2M Fog platform, we carry out a preliminary analysis of different alternative deployments using LTE as a network infrastructure, to analyze the advantages and open issues of each solution. Among them, we propose a deployment that exploits D2D communications between the Fog node and smart objects to enable efficient direct communication between M2M applications and machines. We compare this solution with one based on the legacy LTE configuration and another based on Femtocells, and show, in particular, how the approach based on D2D interactions can guarantee low-latency communication, preserve locality of information, and offer plug-and-play auto-configuration. In order to measure the advantages of each solution, a performance evaluation is carried

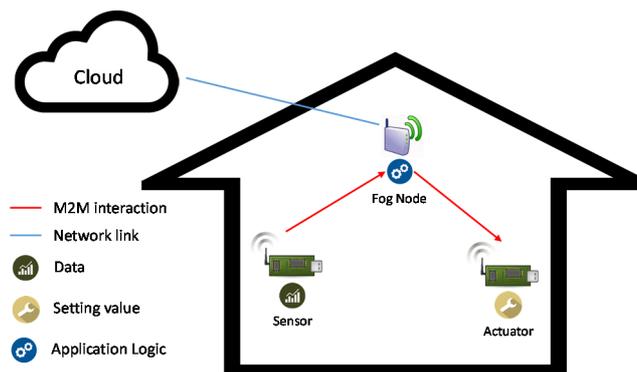


Figure 1 - Fog platform.

out by means of simulations. In particular, we show how M2M applications can rely on D2D communications for real-time low-latency communication with sensors and actuators.

D2D communications have been already proposed as a support for generic Machine-Type Communications (MTC) to reduce the complexity of network management and mitigate the signaling congestion caused by the large number of smart objects connected to the LTE network. In this work, instead, we specifically consider M2M Fog platforms, proposing a practical configuration that supports M2M applications running on Fog nodes to efficiently discover and communicate with local devices, minimizing human intervention in deployment and configuration, required by other solutions based on different technologies, e.g. small-cells.

The remainder of the paper is organized as follows: Section II provides some background information on Fog computing and the LTE standard, Section III presents an analysis of possible deployments, including the proposed approach based on D2D communications. In Section IV we evaluate the performance, and we draw conclusions in Section V, presenting also possible future research directions.

## II. BACKGROUND

### A. Fog Computing

The *Fog computing* paradigm has been recently defined to extend the traditional centralized cloud-based structure towards a distributed architecture that introduces an intermediate layer, called Fog, Figure 1. *Fog nodes* are installed close to physical deployments of smart objects to enable the execution of applications that require low-latency interactions with local devices.

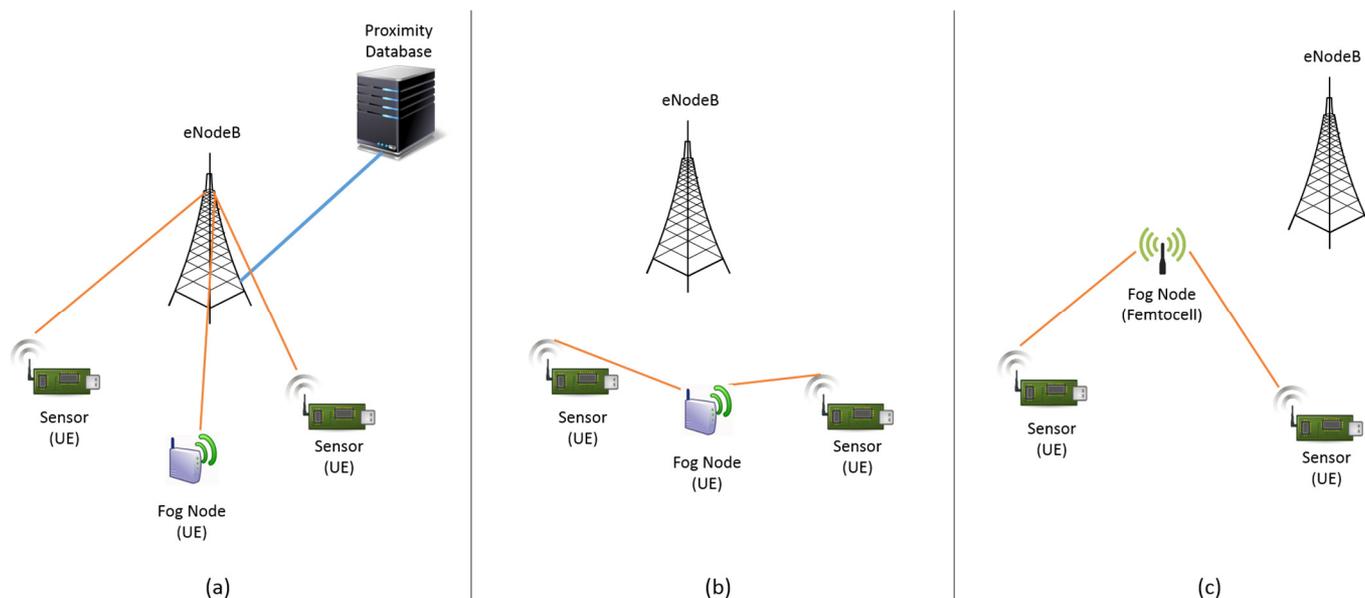
This pervasive architecture is designed to support the execution of M2M applications that implement application logic involving locally deployed smart objects: Fog nodes are installed to support the execution of M2M applications directly among sensors and actuators. The applications can implement some closed-loop control logic, for example to set the value of an actuator based on the readings of a sensor.

Fog platforms can be deployed autonomously, working in isolation, or can be integrated into the centralized cloud as its geographically distributed hierarchical extension. Such integration can be deployed to allow the execution of applications that require higher computational and storage capabilities, not available on Fog nodes, for example for big data processing and historic data collection. Although in their early stage, IoT deployments that adopt the Fog concept in their implementation are available, e.g. the BETaaS platform, [21].

Fog concept is similar to the *Mobile Cloud Computing* [6] architecture, which proposes a distributed architecture co-located within the wireless infrastructure to empower mobile applications with computing and storage capabilities close to mobile devices. Although the overall architectures are similar, their goals and implementations differ: Mobile Cloud Computing platforms aim at providing an infrastructure supporting mobile devices for offloading of applications while Fog platforms are specifically tailored for the future IoT supporting the execution of local applications.

### B. LTE

LTE is the standard for the fourth generation of cellular communications developed by 3GPP. The core part of an LTE system is the Radio Access Network: it is composed of a central evolved-NodeB (eNB) which manages communication within its serving area, called Cell, and one or more User Equipments (UEs), i.e. nodes equipped with an LTE radio interface. Transmissions are arranged in subframes at regular Transmission Time Intervals (TTIs) of 1ms. A group of ten consecutive subframes is called *frame*. ENBs can be of several types, depending on their transmission power and/or cell coverage. High-power nodes are generally used to ensure basic coverage to UEs and are referred to as *macro* nodes. *Small* low-power eNB nodes instead can be used to add localized capacity in



**Figure 2 - Alternative deployments: legacy (a), D2D-based (b), small-cell-based (c)**

hot-spot zones, composing *dense* deployments<sup>1</sup>. As eNBs share the same resources, they interfere with each other, with negative effects on efficiency that increase as the deployment becomes denser. Although these effects can be mitigated via coordination techniques [13], the complexity of the latter in future ultra-dense scenarios is likely to become prohibitive, hence interference can still be expected to hamper the system performance.

Traditional communications are performed in *Device-to-Infrastructure* (D2I) mode, i.e. with the eNB always being one endpoint for a communication. In this case, two peering UEs co-located within the same cell, will communicate through a two-hop path: a first *uplink* (UL) leg from the transmitting UE towards the eNB, and a second *downlink* (DL) leg starting at the latter and terminated at the receiving UE. The eNB is responsible for resource allocation for both the DL and UL directions. However, UEs with backlogged data to transmit must request explicitly UL resources to the eNB by first acceding a random-access shared channel using a backoff procedure, and then sending a Buffer Status Reports (BSR) which signals the eNB the amount of requested resources. Thus, UL communications have higher latency than DL ones.

The eNB can also transmit in broadcast by using Multicast and Broadcast Single-Frequency Network (MBSFN) [9]: some subframes can be declared as MBSFN and used by the eNB to send data to *all* UEs under its coverage. MBSFN subframes do *not* support unicast transmissions, and their number (up to six) and position within a frame must be configured semi-statically [14]. Thus, the eNB has to decide their number and position based on its communication requirements, trading off between unicast and broadcast capacity in a long-term perspective.

In recent years, 3GPP has begun the standardization of a new paradigm of *direct* communication between UEs, called Device-to-device (D2D) [9]. In D2D, peering UEs in proximity are allowed to communicate directly via a *sidelink* (SL), i.e. without using the eNB as a relay. The latter still assists the communication whenever it takes place in licensed resources, thus ensuring a network-controlled D2D communication. The resources needed to ensure communications on the SL are generally allocated on UL frequencies, as UL subframes are expected to be less loaded [19]. In out-of-coverage scenarios instead, the two peering UEs can still communicate *without* network assistance. D2D communication are envisaged to take place in either a one-to-one or a one-to-many fashion [10]: the SL can thus be used for both unicast and multicast communication having UEs as endpoints. A *proximity-discovery* procedure is aimed at finding UEs that are in proximity, thus are suitable for participating to a D2D communication [11]. Proximity can be evaluated according to various criteria, and can be application dependent, location dependent, etc. Note that such flexibility allows for complex discovery procedures that might be based on composited proximity policies, e.g. locality and security.

### III. DEPLOYMENT ANALYSIS

In this section we describe possible solutions to enable Fog computing on LTE networks. We start by describing which network functions are required by the applications. Then, we describe in detail three alternative network deployments and their respective challenges.

<sup>1</sup>Such deployments are often known also as femto-, pico- or micro-cells, and they differ in several aspects, including who is expected to manage them. However, they share the same characteristics of communicating on a shared spectrum in a localized way, thus being complementary, rather than alternative, to the macro.

### A. Network requirements

In order to overcome the current fragmentation of IoT solutions available on the market as proprietary solutions, different protocols and open standards are currently under definition to ensure interoperability with devices from different manufacturers, CoAP, MQTT, etc [18]. Regardless of which specific protocol is used, IoT applications require a minimum set of functionalities from the network to operate:

- Delivery of *unicast* messages for *one-to-one* interactions between applications and smart objects. Such messages are usually exchanged to implement regular operations, such as retrieval of information from a sensor or change of state of an actuator. Applications sensitive to losses and delays require QoS assurances to ensure reliable and timed delivery.
- Delivery of *multicast/broadcast* messages for *local discovery*. Such messages are required to implement self-configuring mechanisms that allow applications to discover devices in their proximity, a functionality that is mandatory to guarantee resilience to changes [8].

Although unicast message delivery, with or without QoS support, is one of the basic functionalities of any network infrastructure, support for multicast/broadcast delivery is less common. If this functionality is not provided natively, the network has to emulate multicast delivery, even though it might be inefficient.

Among the ongoing standardization efforts, the Constrained Application Protocol (CoAP) [7], standardized within a working group of the Internet Engineering Task Force (IETF), is expected to become the standard for communication between smart objects and applications. For this reason and for the sake of concreteness, in the rest of the paper we will adopt CoAP to model the interactions between smart objects and applications to draw general network requirements and to validate our proposal.

CoAP follows the REpresentational State Transfer (REST) paradigm, the same one adopted by the HTTP protocol: smart objects offer their functionalities as server resources, accessed by applications as clients. Request/response transactions implement one-to-one interactions that are issued to perform regular operations. Four different types of methods are defined GET, PUSH, PUT and DELETE, which can be exploited to retrieve measurements from sensors or set values to actuators. In order to provide applications with local discovery functionalities, a service discovery is defined to discover smart objects in their proximity. Such operation is implemented on top of multicast/broadcast transmission.

### B. Analysis

In the following, we offer a critical overview of the possible deployments of an LTE-A network that efficiently supports Fog platforms. Again for the sake of concreteness – and bearing in mind that Fog platforms can be exploited in different contexts, ranging from industrial automation to smart cities - in the following we make reference to a smart home deployment as a use case.

Let us consider a smart home where different sensors, e.g. environmental, intrusion detection, power meter, and actuators, e.g. locks, switches, valves, coexist. These devices are manufactured by different industries and implement a common application protocol, e.g. CoAP. In order to facilitate their deployment and installation, they are equipped with a radio interface to connect to the LTE infrastructure, which provides low-cost connectivity without requiring explicit configuration. Auto-configuration is of paramount importance for IoT devices both to guarantee scalability by excluding human intervention, and – especially in this case - to allow plug-and-play installation by non-technical customers. A Fog platform that supports implementation of application logic based on access to local smart objects, e.g. an intrusion detection application, or an energy management application, is installed through a Fog node.

Figure 2 illustrates three possible deployment alternatives to connect Fog nodes and sensors to the LTE network, namely a *legacy*, a *D2D-based* and a *small-cell-based* deployment.

In the *legacy deployment*, Figure 2 (a), Fog node and sensors connect to the LTE network as UEs associated to the same macro eNB. Communication between UEs (sensor-to-sensor and sensor-to-Fog) is performed in a D2I manner, i.e. data flows through the eNB regardless of the position of the involved UEs. The main advantage of this solution is that both the smart objects and the Fog node are plug-and-play and operator-independent: UEs automatically connect to the network without requiring special configuration or customized deployment from network operators. Such deployment, however, has two main drawbacks: first it makes an inefficient use of *network resources*, since communication always involves the eNB, even when UEs are in direct range. Such inefficient communication also increases the *minimum latency* that can be guaranteed to applications, equal to the round-trip time between the UE and the eNB. Second, it may suffer from loss of *locality information*: considering that a single eNB may cover large areas, one single cell can connect many smart objects, which can be physically distant and even belong to different environments. This drawback not only limits M2M applications that require locality information, but also hinders the implementation of discovery procedures. Without locality information, native multicast offered by the network infrastructure (e.g. via MBSFN) could discover smart objects only on a very large scale, making it infeasible to discover objects in proximity. For this reason, efficient discovery in such deployment requires the introduction of a proximity database, which can link devices in proximity through additional information provided at the time of installation: for instance, in our use case the customer may be required to register its devices on a web page. Moreover, using MBSFN subframes can have a non-negligible impact on the overall system capacity, as unicast transmissions are inhibited in those subframes.

Having many MBSFN subframes hampers normal unicast transmissions, whereas having too few of them may lead to delay in discovering new devices. An efficient allocation of MBSFN subframes requires henceforth to set a trade-off between the above two issues.

Figure 2 (b), instead, presents the D2D-based deployment in which D2D interactions are enabled, allowing UEs to communicate directly when in proximity. Sensors and the Fog node can exploit unicast D2D transmissions for one-to-one interactions and broadcast/multicast D2D ones for local discovery operations. The classical UL connection can be employed to communicate to outside networks (e.g. other Fog nodes or cloud platforms). D2D communications are network-controlled, i.e. the eNB handles UE requests for D2D interactions, checks their feasibility and, in case, allocates resources to them. Explicit resource reservation guarantees interference-free transmission and enforces application QoS requirements. Such deployment overcomes the drawbacks of the legacy approach and preserves the main advantage: the plug-and-play auto-configuration. Direct communication between sensors and the Fog node avoids unnecessary involvement of the eNB, thus reducing the consumption of network resources and minimizing the communication latency. Moreover, the support for multicast delivery in proximity offers applications native support for local discovery operations.

Finally, in Figure 2 (c) we present the *small-cell-based* deployment, where a small eNB is deployed close to IoT devices, inside the house for example. Following a similar approach already proposed in the field of mobile cloud computing [5], the Fog node is co-located with the small eNB, which is usually provided and installed directly by the network operator. Such deployment brings direct connection close to IoT devices and co-locates applications with the core of the access network. It can be henceforth considered as the best-case scenario, as it guarantees low delay (only one hop is always required to communicate with the Fog node) and preserves locality offering native localized multicast support. Similarly to the D2D deployment, the Fog node and sensors can communicate directly reducing delay and required resources. Installation of the small eNB next to the smart objects allows multicast operations in a strictly localized manner that involves *only* devices in proximity. The main drawback of this solution is represented by the mandatory involvement of the network operator, which has to set-up and install an additional network equipment directly on each IoT site. This operation introduces an additional configuration step for connecting sensors to the small-cell, which can become undesirable for large-scale deployments. Finally, we recall that small-cell deployments are generally *dense*, thus prone to inter-cell interference. A massive use of small cells will even amplify this phenomenon, thus requiring more and more complex solutions for interference coordination.

#### IV. PERFORMANCE EVALUATION

In this section, we evaluate the deployments presented above by means of simulation, using SimuLTE [15] for the purpose. The latter is an LTE system-level simulator written in C++ and developed for the OMNeT++ framework [16]. It models all the relevant protocol layers for the data plane of the LTE radio access network, the main nodes of the Evolved Packet Core (EPC), and offers a realistic channel model. We extended it in order to support network-controlled D2D. SimuLTE is based on the INET model suite, henceforth it can be used to simulate realistic internet applications and protocols such as IPv4, UDP. The CoAP protocol is simulated as communication protocol between sensors and the Fog node.

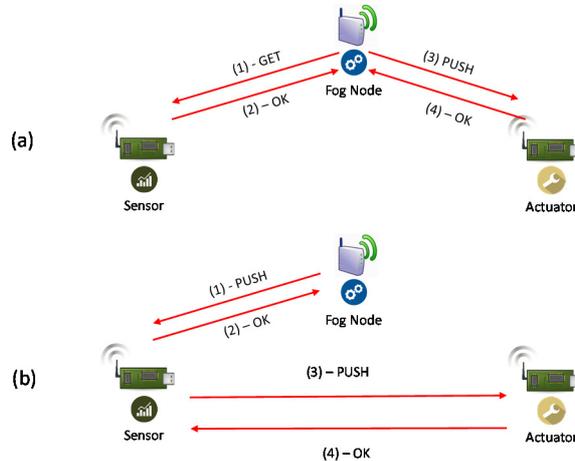


Figure 3 - M2M interactions.

We consider a smart home scenario with one Fog node that runs an application logic and two smart objects, one sensor and one actuator. Two different types of M2M interactions are considered in two simulation scenarios, shown in Figure 3:

1. *Centralized* M2M interactions: the M2M logic is entirely implemented at the Fog node. An example is an application that implements a closed-loop control logic: first a measurement is retrieved from the sensor, then a value is set to the actuator, Figure 3 (a). An example is that the M2M application behaves as a CoAP client, while smart objects expose their functionalities

as CoAP servers. In particular, the sensor provides a measurement in response to a GET invocation, while the actuator triggers its action in response to the invocation of a PUSH method.

2. *Direct* M2M interactions: part of the M2M application logic is implemented on the smart object, which exploits direct interactions with other things to implement autonomous functionalities under the coordination of the application running on the Fog node, Figure 3 (b). An example is an M2M logic implemented on a sensor that triggers an actuator under certain conditions if enabled by the application running on the Fog node, e.g. a presence sensor that triggers an acoustic alarm when presence is detected only if the central alarm running on the Fog nodes is enabled. In this case, the sensor implements a CoAP server to receive the input from the Fog node through a PUSH invocation, e.g. to retrieve a setting, then it triggers the actuator through a PUSH invocation.

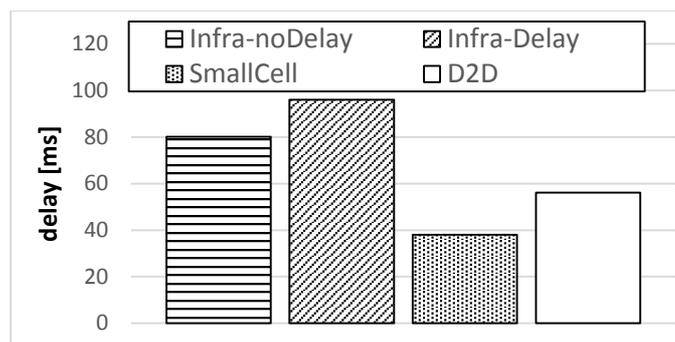
We assume that both local- and proximity-discovery procedures are completed before the simulation starts. We thus simulate and evaluate only the communication phase through the CoAP protocol. The performance of the system is assessed in terms of application *delay*, defined as the time between the request and the response. We consider the three system deployments presented in Section III, plus a fourth one that employs a D2I configuration and counts in also the effects of EPC traversal. In fact, assuming that two-hop D2I communications only traverse the eNB is likely to represent an optimistic scenario as far as delay is concerned. A summary of the simulation parameters is given in Table 1. Confidence intervals are not reported since they are negligible.

First, in Figure 4 we compare the average application delays obtained with the four configurations in a centralized scenario. As we can see, the two *infrastructure* configurations perform the worst, as in both cases the communication between each sensor and the fog node always requires two hops, one in the UL and one in the DL direction. The *D2D* configuration performs better than both the above, but still worse than the *SmallCell* case. Although the number of hops is the same in the two cases, in the latter the communication from the fog node to a sensor is always performed *in the DL*. On the other hand, D2D communications are scheduled on request from the UE, hence involve an additional handshake, which accounts for the increased latency.

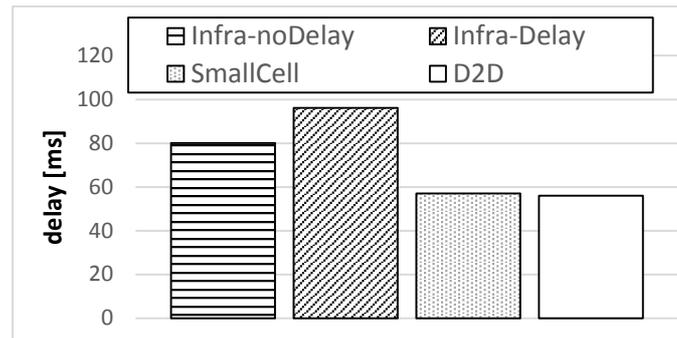
In a *direct* scenario, instead, shown in Figure 5, the *D2D* and the *SmallCell* configurations have similar performance. When using D2D in fact, we are able to curb the number of hops required for the sensor-to-sensor communication, thus reducing the overhead of the UL handshake.

**Table 1 – Simulation Parameters**

Parameter	Value
Carrier Frequency	2 GHz
Bandwidth	5 MHz
Number of RBs	25
Mobility Type	Static
Path Loss Model	ITU Urban Macro/Micro
eNB Tx Power	40 dBm (macro), 30 dBm (micro)
EPC Delay	2 ms
Simulation Duration	60 seconds
# independent replicas	3



**Figure 4 - Comparison of the average application delays in a centralized scenario in various configurations.**



**Figure 5 - Comparison of the average application delays in a direct scenario in various configurations.**

## V. CONCLUSION

In this paper, we discussed how to configure an LTE network to support future M2M Fog platforms. We exemplified a Fog platform taking CoAP as an example and focusing on one-to-one communications and local-discovery operations. In this context, a preliminary evaluation of three different deployments was carried out, discussing in particular the pros and cons of each solution. In particular, we showed that the solution based on network-controlled D2D communications between Fog node and sensors ensures plug-and-play configuration similarly to the legacy solution based on macro-cells, achieving communication performance similar to the solution based on small-cells. As future work, we plan to investigate the scalability of the solution based on D2D communications in large-scale deployments, investigating in particular the overhead of control operations and possible algorithms for ad-hoc resource allocation algorithms.

## ACKNOWLEDGMENT

This work has been partially supported by the project "Metodologie e Tecnologie per lo Sviluppo di Servizi Informatici per le Smart Cities", funded by "Progetti di Ricerca di Ateneo - PRA 2015" of the University of Pisa.

## REFERENCES

- [1] F. Bonomi, R. Milito, J. Zhu, S. Addepalli. 2012. "Fog computing and its role in the internet of things". In Proceedings of the first edition of the MCC workshop on Mobile cloud computing (MCC '12). ACM, New York, NY, USA, 13-16.
- [2] M.R. Palattella, N. Accettura, X. Vilajosana, T. Watteyne, L.A. Grieco, G. Boggia, M. Dohler, "Standardized Protocol Stack for the Internet of (Important) Things," Communications Surveys & Tutorials, IEEE , vol.15, no.3, pp.1389,1406, Third Quarter 2013
- [3] H. Wirtz, J. R uth, M. Seror, J. .B. Link, and Klaus Wehrle. 2014. Opportunistic interaction in the challenged internet of things. In Proceedings of the 9th ACM MobiCom workshop on Challenged networks (CHANTS '14). ACM, New York, NY, USA, 7-12.
- [4] K. Chen, S. Lien, "Machine-to-machine communications: Technologies and challenges", Ad Hoc Networks, Volume 18, July 2014, Pages 3-23, ISSN 1570-8705.
- [5] T.H. Luan, L. Gao, Z. Li, Y. Xiang, L. Sun, "Fog Computing: Focusing on Mobile Users at the Edge", arXiv:1502.01815v1
- [6] H. T. Dinh, C. Lee, D. Niyato, P. Wang, (2013). "A survey of mobile cloud computing: architecture, applications, and approaches". Wireless communications and mobile computing, 13(18), 1587-1611.
- [7] Z. Shelby, K. Hartke, C. Bormann, "The Constrained Application Protocol (CoAP)". Internet RFCs, RFC 7252 (June 2014)
- [8] S. Cirani, L. Davoli, G. Ferrari, R. Leone, P. Medagliani, M. Picone, L. Veltri, "A Scalable and Self-Configuring Architecture for Service Discovery in the Internet of Things," Internet of Things Journal, IEEE , vol.1, no.5, pp.508,521, Oct. 2014
- [9] 3GPP – TS 36.843 v12.0.1, "Study on LTE Device to Device Proximity Services: Radio aspects (Release 12)", March 2014
- [10] Xingqin L., J. Andrews, A. Ghosh, R. Ratasuk, "An overview of 3GPP device-to-device proximity services," Communications Magazine, IEEE , vol.52, no.4, pp.40,48, April 2014.
- [11] A. Khelil, D. Soldani, "On the suitability of Device-to-Device communications for road traffic safety," Internet of Things (WF-IoT), 2014 IEEE World Forum on , vol., no., pp.224,229, 6-8 March 2014, doi: 10.1109/WF-IoT.2014.6803163
- [12] S. Andreevy, O. Galinina, A. Pyattaev, M. Gerasimenko, T. Tirronen, J. Torsner, J. Sachs, Mischa Dohler, and Y. Koucheryavy, "Understanding the IoT Connectivity Landscape – A Contemporary M2M Radio Technology Roadmap," IEEE Communications Magazine, accepted for publication.
- [13] G. Nardini, G. Stea, A. Virdis, D. Sabella, M. Caretti, "Practical large-scale coordinated scheduling in LTE-Advanced networks", Springer Wireless Networks, to appear (Accepted 27/3/2015), DOI: 10.1007/s11276-015-0948-6
- [14] D. Migliorini, G. Stea, M. Caretti, D. Sabella, "Power-Aware Allocation of MBSFN Subframes Using Discontinuous Cell Transmission in LTE Systems," Vehicular Technology Conference (VTC Fall), 2013 IEEE 78th , vol., no., pp.1,5, 2-5 Sept. 2013.
- [15] A. Virdis; G. Stea; G. Nardini, "SimuLTE - A modular system-level simulator for LTE/LTE-A networks based on OMNeT++," Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH), 2014 International Conference on , vol., no., pp.59,70, 28-30 Aug. 2014
- [16] A. Varga, R. Hornig, (2008), "An overview of the OMNeT++ simulation environment", in Proc. SIMUTools '08, Marseille, France, March 2008
- [17] G. Stea, A. Virdis, "A comprehensive simulation analysis of LTE Discontinuous Reception (DRX)", Elsevier Computer Networks, 73 (2014), pp.22-40, DOI 10.1016/j.comnet.2014.07.014

- [18] Z. Sheng; S. Yang; Y. Yu; A. Vasilakos, J. McCann, K. Leung, "A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities," *Wireless Communications, IEEE* , vol.20, no.6, pp.91,98, December 2013
- [19] C. Yu; O. Tirkkonen, K. Doppler, C. Ribeiro, "On the Performance of Device-to-Device Underlay Communication with Simple Power Control," *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th* , vol., no., pp.1,5, 26-29 April 2009.
- [20] 3GPP – TR 23.888 v12.0.0, "System improvements for Machine-Type Communications (MTC)(Release 12)", March 2013
- [21] E. Mingozzi, G. Tanganelli, C. Vallati, V. Di Gregorio, "An Open Framework for Accessing Things as a Service" in *Proceedings of the 16th International Symposium on Wireless Personal Multimedia Communications(WPMC2013)*, Atlantic City, NJ, USA, June 24-27, 2013