Opportunistic Fog Computing: Feasibility Assessment and Architectural Proposal

Ricardo Silva* [1], Jorge Sá Silva‡ [2] and Fernando Boavida‡ [3]
*Skyline Communications
Ambachtenstraat 33
B-8870 Izegem, BELGIUM
[1] ricardo.silva@skyline.be

‡Department of Informatics Engineering
University of Coimbra, Pólo II - Pinhal de Marrocos, 3030-290 Coimbra, PORTUGAL
[2] sasilva@dei.uc.pt
[3] boavida@dei.uc.pt

***Abstract*—Cloud computing has been pointed out as one of the pillars of the Internet of Things, providing this emerging infrastructure with unlimited resources and easing its deployment. However, cloud computing relies on remote, mainly centralized resource provisioning, which poses substantial problems in the support of time-critical and location-aware IoT applications. Fog computing, on the other hand, recently emerged as an intermediate solution to solve the mentioned cloud computing limitations, but its embryonic status still prevents it from being used in real developments.**

In this paper we present a fog computing characterization and assess the feasibility of its use in a worst-case scenario, which we name opportunistic fog network. Subsequently, as a first step towards its effective use and deployment in an IoT scenario, we propose a fog computing network architecture, constituted by virtual clusters running on peer-to-peer overlays, capable of abstracting the complexity of lower layers.

Opportunistic fog computing can benefit from thousands of end-devices moving in urban areas everyday, providing connectivity and processing resources to every crowded spot, and promoting green computing by allowing a more efficient local resource usage and by decreasing remote communications to the cloud. Furthermore, opportunistic fog computing overcomes cloud computing drawbacks whenever a set of independent and heterogeneous end-devices agree to share their own resources.

***Index Terms*—Fog computing, Virtual cluster, resource-sharing, Internet of Things.**

I. Introduction

At the beginning of this century, tiny, resource-constrained, battery-powered, sensor-enabled devices emerged and proliferated. As stated by Friedemann Mattern [1] in 2004 the "everyday objects became smart and networked". Nowadays, dealing with an enormous and still increasing amount of such "everyday objects" requires a gigantic backend supported by large datacenters capable of handling big data, with dynamic resource allocation. Cloud computing has become a fundamental piece of the Internet of Things deployment, providing a reliable and powerful interface between heterogeneous smart objects and the Internet. However, because cloud computing operates in a centralised fashion, it is unable to respond adequately to high mobility, low latency or location awareness requirements [2]. Hence, considering the numerous types of devices that surround us nowadays, Cisco introduced a new concept capable of overcoming the limitations of cloud computing for IoT support: fog computing [3]. Fog computing operates at the edge network, right between the smart objects and the cloud computing infrastructure, exploring physical proximity and, consequently, enabling quicker response. As mentioned by Mung Chiang [4] there are four reasons to implement fog computing: i) Time-Real time applications and Cyber-Physical Systems require milliseconds responses; ii) Cognition - objectives awareness; iii) Efficiency - making use of locally available resources; iv) Agility - possibility of adopting and experimenting new setups faster, without having to wait for their availability from big vendors.

Fog computing requires the cooperation of edge devices, such as routers, set-top boxes and end-user devices (smartphones and tablets, among others) to build a platform capable of guaranteeing the same levels of high availability (HA) existing in cloud computing, while solving its limitations. Resource sharing among heterogeneous devices poses several challenges, which we have previously identified in [5]. The coexistence of devices is not enough to guarantee cooperation among them, leading to technical and non-technical issues. Technical issues concern physical incompatibilities among devices, requiring the establishment of standards and agreements. On the other hand, at non-technical level, owners have to agree on lending/sharing their hardware, possibly supporting symbiotic operation.

Fog computing appeared inline with our previous work [6], [7], in which we proposed the Network of Proxies (NoP) concept, a network constituted by non-or-less-constrained nodes capable of assisting resource-constrained devices in performing computationally or energetically heavy tasks on their behalf. However, implementing NoP and fog computing on edge devices poses several challenges. While NoP was developed for critical and controlled scenarios, fog computing is generic and should be implemented by any edge device. Edge devices are usually individually managed, may be mobile, and are

owned by different people, whose security and privacy concerns vary. Hence, a fog computing implementation must a) be capable of guaranteeing high availability of services regardless the dynamic of the supporting edge devices, b) be able to guarantee security and privacy regardless the supporting edge devices, and c) assure a more effective support of time-demanding and critical IoT applications when compared to cloud computing.

We can thus define fog computing as an intermediate service existing between objects and cloud computing, capable of supporting immediate or critical requests on behalf of cloud computing, while assuring location-awareness, mobility, security and privacy.

The main contributions of this paper are: (1) a characterization of fog networks regarding the application scenario; (2) a fog computing feasibility assessment, considering the worst case scenario (opportunistic fog) defined in (1); and (3) the proposal of a fog computing architecture based on peer-to-peer (P2P) and virtual clustering, capable of efficiently supporting the previously defined scenarios.

The remaining of this paper is organized as follows. Section II provides the background in this area. Section III presents a fog network characterization, while section IV assesses the feasibility of the worst case scenario identified in III. Section V presents the proposed architecture, and Section VI summarizes the findings and outlines future work.

II. Background

Cisco implemented and made available the fog computing concept through their recent product IOX [8]. IOX mixes Cisco's operating system IOS with Linux, enabling resource sharing among Cisco devices, such as, routers and switches. Nevertheless, the concept of fog computing is wider, and extending it to generic devices is not simple, requiring a common middleware capable of managing resource sharing and abstracting platform complexity, while providing computing services.

In [9] authors presented a cloudlet solution in which generic edge devices are associated to one virtual platform implemented via Openstack++ . Cloudlet is considered a rich-resource fog node, built on a three-layer architecture with a Linux base, a virtualization layer, and a set of independent virtual machines running on top. Cloudlet prototypes can be easily included within the fog computing concept, being perhaps the closest open implementation.

In [10] authors presented a mathematical framework for resource sharing, based on a service-oriented function, which, considering the heterogeneous resources available as a "time unit", demonstrated a reduction on service latencies and energy consumption when compared to cloud computing. In [11] authors formulated an optimal workload allocation between fog and cloud, targeting minimal power consumption, demonstrating that using fog computing on constrained edge devices saves communication bandwidth and improves the overall performance. However, as demonstrated in [12], fog computing outperforms cloud computing only when considering a high number of latency-sensitive applications in the IoT context (specifically when latency-sensitive applications generate at least 50% of requests). Otherwise fog computing becomes an additional overhead when compared to traditional cloud computing. This result is important to frame fog computing within the application scenarios and their demands, but it is not absolute because fog computing can be applied not only to decrease latencies but also to improve applications by providing additional features and resources, such as applications offloading and storage, as proposed in [13].

Many other publications present, survey and characterize fog computing [14], [15], [16], [17]. However, as this is quite a recent area, the state of the art is still incipient regarding architecture definition, middleware and real implementations. Nevertheless, some consortiums are attempting to standardize the fog architecture, with OpenFog [18] taking the lead.

Given the state of the art presented above, it is important at this stage to characterize potential fog computing scenarios, assess their technical feasibility, and propose an architecture that can be effectively used in real IoT deployments. This is done in the remaining sections of this paper.

III. Fog Network Characterization

The implementation of the fog computing concept poses several challenges, as previously mentioned in the introduction of this paper. The very first challenge is the network architecture. Different types of fog network organization can be considered, depending on the device cooperation scope and scenario. We propose the following characterization of fog network architectures:

- Home fog - A local network constituted by a small set of static and few mobile nodes. Fog computing may be established using, for instance, a few nodes existing at home, such as, set-top boxes, smart-TV, routers, switches, smartphones and even personal computers.
- Building fog - A network constituted by tenths of static and mobile nodes typically existing within buildings, such as, servers, routers, switches, personal computers, tablets, smartphones, hotspots, among others. These edge devices may be owned and managed by different people.
- Enterprise fog - A network constituted by tenths to a few hundred static nodes, typically owned by the same entity. Typically, mobile edge nodes, such as, smartphones are not included in Enterprise Fog, unless they are owned and managed by the enterprise.
- Opportunistic fog - a network organically created by a set of mobile devices, such as, smartphones and tablets sharing the same physical area at the same time.

Home fog is the basic and consequently less problematic fog network. Nowadays people maintain several pieces of electronic equipment at home, most of them connected 24/7. The best examples are set-top boxes, routers and switches. Although these edge devices have their own specific purpose, their processing, storage and communications resources are far from being used to their fullest capacity, wasting considerable energy. On the other hand, houses are becoming smarter

and full of sensors and actuators deployed in the scope of emerging IoT applications. Those applications usually resort to remote cloud computing services to deliver their services. For instance, popular IP video cameras, when deployed, automatically provide video streaming via cloud. The same occurs with power meters, windows and ambient controllers, among others. Most of the time people are using those IoT applications right at home, while all personal information is travelling to a remote cloud-based server and the responses travel back home again. In addition to posing security and privacy issues, this mode of operation leads to waste of resources, namely the resources spent in the cloud servers as well as networking resources such as bandwidth, routers, and switches processing capacity. With the implementation of home fog, local resources can be used more efficiently and many IoT applications can be run locally, improving quality of service, privacy and security.

Building fog represents a wider scope network when compared to home fog, and the goal is to profit from many of the edge devices available in a building or a set of adjacent buildings. Nowadays buildings are equipped with sets of active network equipment that can be availed to setup an effective fog network. Apartment or office buildings, among others, can hold a fog computing set of services capable to improve users experience, with lower latencies, mobility support, location-awareness, security and privacy while running IoT applications. Building fog introduces additional challenges when compared to home fog, namely the number of edge devices available and the fact that those devices may be owned and managed by different people, who can also allow the participation of their own mobile devices, such as smartphones and tablets. This sharing of resources will enable more powerful fog networks right when the need is higher, due to the fact that more users mean more devices (which are normally under-utilized) to support the various applications and networking needs.

Enterprise fog is similar to building fog in the sense of network dimension, but different in terms critical requirements and network management. In general, all edge devices existing in a company are owned and managed by the same entity, easing the fog implementation when compared to building fog. However, unlike building fog, enterprise fog will be, in general, used to support IoT applications of higher criticality, which leads to stricter requirements in terms of availability, response-time, security and privacy, among other quality of service requirements.

Opportunistic fog is the most challenging type of fog computing architecture and the most hard to setup and manage. Opportunistic fog relies on a potentially high number of mobile devices, such as smartphones and tablets that nowadays share our ecosystem. Although these mobile devices may have some individual resource constrains, together they represent a high amount of processing power, storage and bandwidth. Opportunistic fog is a fog organization supported by mobile edge devices sharing the same space at the same time. It can be established in public spaces, such as train, metro or bus stations, coffee shops, restaurants, cinemas, museums and any crowded space. Opportunistic fog has enormous potential to deliver IoT applications with higher quality of service, guaranteeing low latencies and full mobility support. Since the fog network will use heterogeneous devices owned by different people, it poses several challenges concerning its establishment and maintenance, as well as security and privacy concerns. Whenever opportunistic fog networks can take advantage of static edge devices available and willing to share resources, they can be extended, increasing their capacity and improving stability. Train, metro and bus stations are scenarios where this can be a reality, with opportunistic fog networks being enhanced by local infrastructure. An interesting use-case would be the real-time support in panic situations (e.g.: in train stations). In such scenario, opportunistic fog networks could be used to hold an application use-case to provide local assistance, such as, redirecting people to safe areas, providing useful information in real-time and calling emergency services whenever needed. Table I summarizes the presented characterization.

IV. Opportunistic Fog Feasibility Assessment

Opportunistic fog occurs whenever a set of participants is in the same area at the same time, sharing their computational resources. Guaranteeing quality of service over opportunistic platforms is a major challenge that faces many uncontrolled variables, with participants' mobility on top.

The purpose of the work presented in this section was to assess the feasibility of opportunistic fog, as well as to anticipate its behavior under different usage conditions such as node density and mobility models. To do so, we simulated several scenarios in the Cooja [19] simulator with the immediate objective of determining the evolution of neighborhood density and, subsequently, of inferring the viability of opportunistically building a fog network. Although Cooja was designed to simulate wireless sensor networks and not fog networks, the support of multi platforms and protocols with mobility provision makes it an excellent platform to carry out the intended study.

In this evaluation we aimed at simulating the natural behavior of edge devices in a public, urban area, such as, metro and train stations, shopping centers, or cinemas, where the concentration of computational resources is typically high, even though only during specific periods of the day. The coexistence of such resources has the potential to enable the establishment of an opportunistic fog network that can complement a cloud network and assist service provisioning.

The success of the opportunistic fog network directly depends on the available resources, i.e., on the number of nodes available for sharing their resources and participating in the fog establishment. Hence, at this earlier stage, we found it important to analyze the feasibility of opportunistically building fog networks in highly mobile scenarios, analyzing the neighborhood density along time while varying the mobility

TABLE I
SUMMARY OF FOG NETWORK CHARACTERIZATION.

	Mobility	Critical Level	Pros	Cons
Home	Low	Low	Resource exploitation, security, privacy, latency	Limited
Building	Medium	Medium	Resource exploitation, latency	Edge devices managed by different people
Enterprise	Low	High	Resource exploitation, latency, security, privacy	Limited
Opportunistic	High	Low	Resource exploitation, improving users experience everywhere	Node discovery, Network management

pattern of fog participants. Considering the example of train stations, people are dynamically arriving at and leaving the facilities, which means that the available edge devices, such as smartphones and tablets, keep changing. To simulate this environment, we considered a square of 100x100 meters and a fog network of up to 25 mobile nodes. We also considered that each node has a radio range equal to 20 meters. Nodes were running an out-of-the-box UDP client-server application on top of 6LoWPAN with RPL - the de facto standard for the Internet of Things. This setup was designed considering also the computational resources available. Simulating 25 nodes randomly moving requires high processing capacity. Fig. 1 depicts a screenshot taken from one of the simulations.

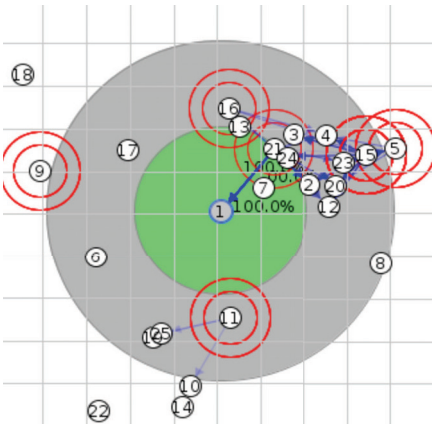

Fig. 1. **Simulation scenario in Cooja.**

Cooja was enhanced with the mobility plugin, and bonnmotion [20] was used to build the mobility trace files, considering the following mobility models: random walk, probabilistic random walk, gauss-markov, reference point group mobility (RPGM).

These models were chosen based on their characteristics and their proximity to the most typical scenario in which opportunistic fog networks will occur. In the random walk model nodes randomly change direction and speed, while in the probabilistic random walk the model uses a probability table to decide the next move. The gauss-markov model [21] models the node velocity over time as a Gauss-Markov stochastic process, and the reference point group mobility (RPGM) [22] models group mobility in which the group members follow the group leader, modelling soldiers in field, rescue scenario or just people entering or leaving public transports.

For each model we simulated the networks operation during the period of one hour, measuring the node relationship ratio, i.e., the neighborhood density (neighbors count) along time. The higher the density the more feasible will be the constitution and maintenance of the opportunistic fog network. In this scenario, for practical reasons, we limited neighborhood density to a maximum of 20 nodes, which represents 80% of all the nodes available in the area, considering it the highest density under the given assumption.

Fig. 2 presents the obtained results considering the four mobility models. In this figure, each of the 25 nodes is represented by a different color/shape spot, as indicated in the legend. For instance, node 6 is represented by a brownish circle, while node 11 is represented by a light blue square. As we can observe, in all cases the neighborhood density increases over the first minutes until it reaches the maximum of 20 nodes. In the cases of the RPGM and random walk models the neighborhood density rapidly reaches the maximum value, while the probabilistic random walk and gauss-markov models take more time to reach the densest neighborhood level, with the former presenting the less stable and less dense neighborhood. Nevertheless, we can conclude that in the first ten minutes, which also includes the bootstrap overhead, nodes can get from an average of five to ten neighbors, increasing this number to close to 20 neighbors in the next few minutes. The obtained results point to the existence of dense neighborhoods under random mobility, thus supporting the theses of the feasibility of opportunistic fog connectivity of mobile devices sharing the same area at the same time in urban scenarios, such as the one under consideration in the performed simulations.

Nevertheless, a logical infrastructure must be established on top of the opportunistic network infrastructure, regardless its constitution, guaranteeing the establishment of an effective fog computing platform, seen as a coherent resource for connectivity and computation. This is the main rationale for the architecture proposal presented in the next section.

V. WHERE PEER-TO-PEER MEETS VIRTUAL CLUSTERING

The characteristics and limitations of cloud computing are pushing forward new paradigms for building the Future Internet, such as the fog computing paradigm. Lower latencies, mobility support, location awareness, better security and privacy are just a few advantages that fog computing is supposed to bring. An old argument used by peer-to-peer (P2P) and content distribution networks (CDN), mentioned in [18], states that proximity is in the edge, which means that it is more efficient

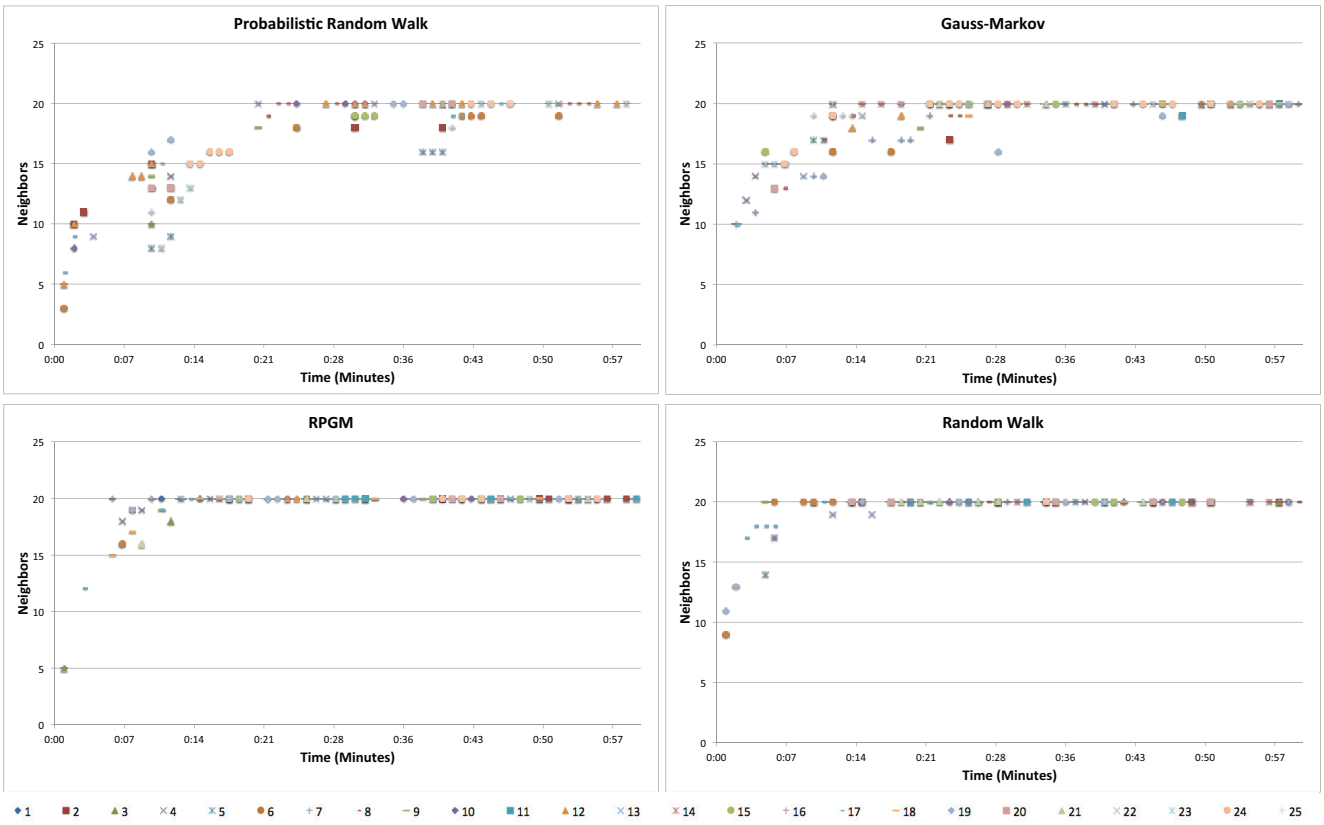

Fig. 2. **Neighborhood density on different mobility models.**

to communicate and distribute data among Fog computing is characterized by the following:

- Fog gets closer to the phenomena - Things and people are at the edge. As a consequence, latencies will be much lower, since communications are made locally;
- Fog eliminates unnecessary network traffic - the fog is built at the neighborhood level and therefore the communication to cloud, when it occurs, is filtered;
- Fog guarantees a more restrictive environment - sending personal data to the edge is quite different from sending data to a remote centralized service. Implementing security and privacy mechanisms at the edge is simpler and more effective.
- Fog supports mobility and location-awareness applications - at the edge is possible to closely monitor objects and know whether they are in the range or not, as we previously demonstrated with NoP [6].
- Fog goes green - By reducing remote communications and taking advantage of unused capacity in local devices, fog promotes a more efficient use of resources.

Nevertheless, fog implementation is not trivial and many issues still need to be defined, implemented and standardized. As mentioned in Section III, fog networks may be constituted in different scenarios, corresponding to different levels of complexity. Although opportunistic fog is by far the most complex scenario, the remaining scenarios may also pose considerable challenges due to the inclusion of mobile edge devices. As a consequence, the fog network should be able to quickly deal with the dynamics of heterogeneous edge nodes, which may be entering or leaving the area, maintaining service availability with good performance. In light of this, we propose the three-level architecture presented in Fig. 3, detailed in the next subsection.

A. *Fog computing architecture*

As presented in Fig. 3 the proposed architecture relies on three layers: the network layer (layer 1), the system layer (layer 2) and the services layer (layer 3). The network layer should allow the inclusion of all edge devices willing to contribute to the fog computing environment, guaranteeing stability, performance and quality of service. The system layer must abstract the complexity of the network by providing a single-system image to the next layer. The services layer can include any fog service, deployable in any platform, and totally unaware about what is happening in lower layers.

Hence, inline with [15] and [18], we propose the use of an unstructured, peer-to-peer network architecture to act as the basis of the proposed fog computing architecture (Layer 1). By using a peer-to-peer approach, network management gets decentralized and can follow several existent solutions, such as OMAN [23].

Regardless the fog network variations, quality of service

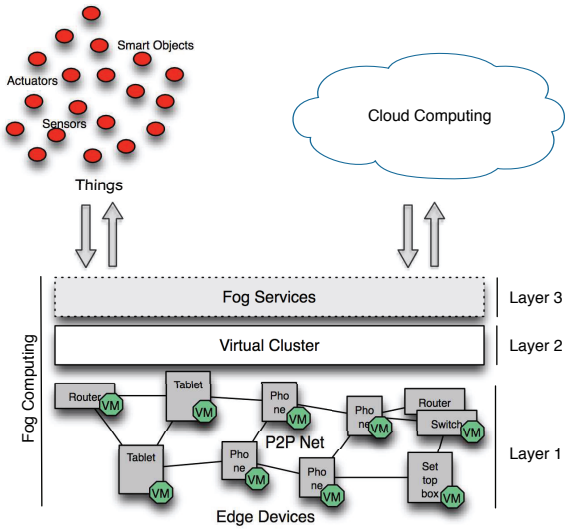

Fig. 3. **Proposed fog computing architecture.**

should not be affected nor the services limited. Besides, the network architecture must be completely abstracted and the supported services totally independent from it. To do so, we propose the use of virtualization, such as in cloud computing and inline with [3] and [16]. Virtualization brings the desired abstraction while providing nodes isolation and improving security and privacy. Furthermore, virtual machines can be migrated among nodes considering their available resources, such as processing capacity, storage, bandwidth, energy (whenever dealing with mobile edge nodes), and distance to the neighborhood.

On top of the P2P network, we propose to use a virtual cluster (Layer 2) composed by a set of virtual machines running on edge devices. Those virtual machines should be light and host-based. The use of a virtual cluster allows, as mentioned before, live migration of VM, memory and files, and also dynamic deployment.

With a reliable virtual cluster implemented on top of a P2P network it is then possible to support any type of fog service with elastic resources, which, even though not comparable to existing services at cloud level, are enough to cover many needs of end users, especially at local level. The virtual cluster should be managed to deliver the following benefits [24]: scalability, while maintaining adequate performance; high availability (HA); and, fault tolerance.

Although fog computing is designed to be placed between things/objects and cloud computing and, therefore, it is not intended to support large processing and storage capacity, performance scalability may be needed mainly for supporting demanding real time applications, such as video streaming.

B. Implementation strategy

To implement the proposed architecture, we aim to develop easy-to-deploy and ease-to-use apps for both mobile and static devices. These apps include not only typical mobile applications to run on Android or iOS, but also plugins to enable the most usual set-top boxes and home routers to participate in the fog network. The app allocates local resources to a generic virtual machine that can be migrated from the neighborhood or created as new. The amount of locally allocated resources can be defined by user or dynamically adjusted regarding the current state of the supporting device. Most of the supporting hardware would be mobile and therefore power consumption and energy management must be a primary concern. Also, security and privacy must be guaranteed through the total abstraction between the hardware supporting the virtual cluster and the content running on top of it. Apps and plugins will be freely distributed, and source code will be freely available, promoting the establishment of and participation in an open community.

VI. Conclusions and Future work

Cloud computing has limitations, mainly in what concerns the support of time-demanding and location-aware applications, in addition to mobility, security and privacy issues. On the other hand, the proliferation of billions of edge devices, such as smartphones, tablets, set-top boxes and routers, among others, offers an enormous unused capacity of storage, processing and networking.

In this paper we approached fog computing following a bottom-up perspective. After characterizing fog computing environments, we came up with the term of opportunistic fog as a worst case scenario, in which fog computing is organically established over edge-devices available at a given moment in a given area. Due to its random nature, we found it necessary to assess and prove the feasibility of such a concept, even in sparse user environments. Through simulation of an urban area of 100x100m with only 25 nodes and using a variety of mobility patterns, we were able to conclude that the number of connected neighbors can be quite high, and connectivity can be achieved with a small number of hops. In addition, we proposed an innovative architecture, based on peer-to-peer and virtual clustering, for use in all of the characterized types of fog networks, including opportunistic fogs.

As ongoing work we are implementing and subsequently will assess the proposed architecture on a real platform. As mentioned in the previous section, we also aim to release a generic app to be installed on mobile edge devices, promoting the proliferation of opportunistic fog networks.

Acknowledgment

The work presented in this paper was partially carried out in the scope of SOCIALITE Project (PTDC/EEI-SCR/2072/2014), co-financed by COMPETE 2020, Portugal 2020 - Operational Program for Competitiveness and Internationalization (POCI), European Unions ERDF (European Regional Development Fund), and the Portuguese Foundation for Science and Technology (FCT).

REFERENCES

[1] F. Mattern and E. Zrich, "Wireless future: Ubiquitous computing," in *In: Proceedings of Wireless Congress 2004*, 2004.

[2] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, ser. MCC '12. New York, NY, USA: ACM, 2012, pp. 13–16. [Online]. Available: http://doi.acm.org/10.1145/2342509.2342513

[3] CISCO, "Fog computing and the internet of things: Extend the cloud to where the things are," https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-overview.pdf, Jan 2017.

[4] M. Chiang and T. Zhang, "Fog and iot: An overview of research opportunities," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–1, 2016.

[5] R. Silva, J. S. Silva, and F. Boavida, "A symbiotic resources sharing iot platform in the smart cities context," in *2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, April 2015, pp. 1–6.

[6] ——, "A proposal for proxy-based mobility in {WSNs}," *Computer Communications*, vol. 35, no. 10, pp. 1200 – 1216, 2012.

[7] ——, "Mobility in wireless sensor networks - survey and proposal," *Computer Communications*, vol. 52, no. 0, pp. 1 – 20, 2014.

[8] CISCO, "Cisco fog computing solutions: Unleash the power of the internet of things," https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-solutions.pdf, Jan 2017.

[9] M. Satyanarayanan, R. Schuster, M. Ebling, G. Fettweis, H. Flinck, K. Joshi, and K. Sabnani, "An open ecosystem for mobile-cloud convergence," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 63–70, March 2015.

[10] T. Nishio, R. Shinkuma, T. Takahashi, and N. B. Mandayam, "Service-oriented heterogeneous resource sharing for optimizing service latency in mobile cloud," in *Proceedings of the First International Workshop on Mobile Cloud Computing & Networking*, ser. MobileCloud '13. New York, NY, USA: ACM, 2013, pp. 19–26. [Online]. Available: http://doi.acm.org/10.1145/2492348.2492354

[11] R. Deng, R. Lu, C. Lai, T. H. Luan, and H. Liang, "Optimal workload allocation in fog-cloud computing towards balanced delay and power consumption," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–1, 2016.

[12] S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the suitability of fog computing in the context of internet of things," *IEEE Transactions on Cloud Computing*, vol. PP, no. 99, pp. 1–1, 2015.

[13] M. A. Hassan, M. Xiao, Q. Wei, and S. Chen, "Help your mobile applications with fog computing," in *2015 12th Annual IEEE International Conference on Sensing, Communication, and Networking - Workshops (SECON Workshops)*, June 2015, pp. 1–6.

[14] S. Yi, C. Li, and Q. Li, "A survey of fog computing: Concepts, applications and issues," in *Proceedings of the 2015 Workshop on Mobile Big Data*, ser. Mobidata '15. New York, NY, USA: ACM, 2015, pp. 37–42. [Online]. Available: http://doi.acm.org/10.1145/2757384.2757397

[15] L. M. Vaquero and L. Rodero-Merino, "Finding your way in the fog: Towards a comprehensive definition of fog computing," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 5, pp. 27–32, Oct. 2014. [Online]. Available: http://doi.acm.org/10.1145/2677046.2677052

[16] S. Yi, Z. Hao, Z. Qin, and Q. Li, "Fog computing: Platform and applications," in *2015 Third IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*, Nov 2015, pp. 73–78.

[17] M. Yannuzzi, R. Milito, R. Serral-Graci, D. Montero, and M. Nemirovsky, "Key ingredients in an iot recipe: Fog computing, cloud computing, and more fog computing," in *2014 IEEE 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, Dec 2014, pp. 325–329.

[18] OpenFog, "Openfog architecture overview: Openfog consortium architecture working group," www.openfogconsortium.com, Jan 2017.

[19] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, "Cross-level sensor network simulation with cooja," in *Proceedings. 2006 31st IEEE Conference on Local Computer Networks*, Nov 2006, pp. 641–648.

[20] "The bonnmotion website," https://sys.cs.uos.de/bonnmotion, Jan 2017.

[21] B. Liang and Z. J. Haas, "Predictive distance-based mobility management for pcs networks," in *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, Mar 1999, pp. 1377–1384 vol.3.

[22] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, "A group mobility model for ad hoc wireless networks," in *Proceedings of the 2Nd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '99. New York, NY, USA: ACM, 1999, pp. 53–60. [Online]. Available: http://doi.acm.org/10.1145/313237.313248

[23] A. Fiorese, P. Simões, and F. Boavida, *OMAN – A Management Architecture for P2P Service Overlay Networks*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 14–25. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-13986-4_3

[24] K. Hwang, J. Dongarra, and G. C. Fox, *Distributed and Cloud Computing: From Parallel Processing to the Internet of Things*, 1st ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.