Fog Computing for the Internet of Mobile Things: issues and challenges

Carlo Puliafito§*, Enzo Mingozzi*, Giuseppe Anastasi*

*Department of Information Engineering, University of Pisa, Pisa, Italy
carlo.puliafito@ing.unipi.it, e.mingozzi@iet.unipi.it, giuseppe.anastasi@unipi.it

§DINFO, University of Florence, Florence, Italy
carlo.puliafito@unifi.it

Abstract—**The Internet of Things (IoT) conceives a world where everyday objects are able to join the Internet and exchange data as well as process, store, collect them from the surrounding environment, and actively intervene on it. An unprecedented number of services may be envisioned by exploiting the Internet of Things. Fog Computing, which is also known as Edge Computing, was proposed in 2012 as the ideal paradigm to support the resource-constrained IoT devices in data processing and information delivery. Indeed, the Fog, which does not replace the centralized Cloud but cooperates with it, distributes Cloud Computing technologies and principles anywhere along the Cloud-to-Things continuum and particularly at the network edge. The Fog proximity to the IoT devices allows for several advantages that must be continuously guaranteed, also when end devices move from one place to another. In this paper, we aim at examining in depth what it means to provide mobility support in a Fog environment and at investigating what are the main challenges to be faced. Besides, in order to highlight the importance of this topic in everyday life, we provide the reader with three scenarios where there is an integration between the IoT and Fog Computing, and in which mobility support is essential. We finally point out the main research directions in the field.**

Keywords—Smart Cities; Internet of Things; Fog Computing; Edge Computing; Mobility Support; Service Migration.

I. INTRODUCTION

The Internet of Things (IoT) has proved to be a real revolution in Information and Communication Technologies (ICT). It conceives a world in which every single object, from a computer to a not communicating "dumb" thing (e.g., a lamp post, a dumpster, a refrigerator), can join the Internet. Such objects may not only exchange data, but they can also store and process them as well as collect them from the surrounding environment through sensors and actively intervene on it through actuators. Besides, people can take part in this ambitious scenario, consuming and producing data through their smartphones and wearable devices. The IoT is strongly disruptive: with the widespread distribution of these “smart” devices, cities may finally transform into Smart Cities which can be sensed and controlled, and a large number of services may be envisioned for the betterment of citizens’ lives. In order for these services to become real, the great amount of collected data (i.e., the Big Data) needs to be processed and the obtained insights need to be retrieved. However, data processing and information delivery cannot be performed by the IoT devices themselves because of their limited physical resources. To overcome this issue, the IoT devices may offload the collected data and computation onto the Cloud, exploiting its almost limitless resources.

In 2012, Bonomi et al. [1] proposed Fog Computing, which is also known as Edge Computing, as the ideal paradigm to support the resource-constrained IoT devices. Indeed, Fog Computing, which is not supposed to replace the centralized Cloud but to coexist and cooperate with it, distributes Cloud Computing technologies and principles (i.e., virtualization, scalability, pay-per-use model) anywhere along the Cloud-to-Things continuum and particularly at the network edge, in close proximity to the IoT devices. Therefore, rather than always offloading data and computation onto a distant, centralized Cloud, the smart objects may also take advantage of closer, resource-rich, and Cloud-enabled nodes, namely the Fog nodes. Their proximity to the IoT devices is the key enabler of several advantages that were not possible when continuously offloading onto the distant Cloud. The principal ones are:

- a low and predictable latency;
- a substantial reduction of bandwidth consumption;
- more privacy; and
- context awareness.

Fog Computing is a relatively new concept; therefore, it still presents a large number of challenges and open issues to be solved. Some of the most fascinating have to be faced in order to provide mobility support to the end devices: the aforementioned advantages of Fog Computing need to be continuously guaranteed, not only when nodes are static, but also when they move from one place to another. Nodes mobility compromises Fog Computing benefits indeed, as, when a node moves, the distance between it and the Fog node that hosts the Fog application component increases. Therefore, what has to be done is to migrate the Fog application com-

978-1-5090-6517-2/17/$31.00 ©2017 IEEE

ponent from one Fog node to another, keeping it close enough to the associated mobile node. Thus, mobility in a Fog environment raises computational challenges besides the networking ones. In this paper, we investigate the issue of mobility support in a Fog environment, focusing on a specific category of mobile nodes: the mobile IoT devices. In particular, we survey the related work in the field and identify possible new research directions.

The rest of the paper is organized as follows. Section II presents three scenarios where mobility support is essential. Section III provides a description of the problem and of the main challenges, while Section IV reports the related work. In Section V, we point out the main research directions in the field. Finally, in Section VI, we draw some conclusions.

II. SCENARIOS OF MOBILE IOT IN A FOG ENVIRONMENT

The number of mobile devices connected all over the world is everyday increasing at an unprecedented rate. According to [2], global mobile devices were 7.6 billion in 2015, 8.0 billion in 2016 and are predicted to reach 11.6 billion by 2021.

Our focus, though, is on a specific subset of mobile devices: the IoT ones. Mobile IoT is an ever-increasing phenomenon and forecasts are definitely impressive. Reference [2] reports that wearable devices, which are probably the best known example of mobile IoT devices, were 325 million all over the world in 2016 and are expected to become 929 million by 2021. This great amount of mobile IoT devices may take advantage of Fog Computing in order to enable services that can dramatically improve people's lives. We now discuss three scenarios where there is an integration between the IoT and Fog Computing, and in which mobility support is essential.

1) Citizen's healthcare: This scenario is depicted in Fig. 1. One or more IoT devices are exploited to sense a patient’s health-related parameters (e.g., brain activity, heartbeat, breath, position) and perform some actions (e.g., send a mild electric shock, make an emergency call). All these devices are embedded in, or wirelessly connected to, the patient's smartphone, which always runs the frontend application component. The backend component is deployed at the Fog layer instead. It receives the collected data and performs resource-intensive computations in order to extract valuable information from them and determine the actions to be performed. The promptness through which these actions are triggered is of paramount importance and can often make the difference between life and death. Furthermore, the collected data are usually personal, and thus privacy has to be guaranteed. Therefore, it is vital to keep the Fog application component close enough to the patient as she moves throughout the city. At the beginning of the considered scenario, the patient is at home, and the Fog service (i.e., the orange circle in figure) is hosted on the powerful Wi-Fi access point in the patient's house. She then goes out and gets across town to reach the hospital for a checkup. As the patient moves, the Fog application component has to be migrated from one Fog node to another in order to continuously ensure the aforementioned requirements. Fig. 1 reports typical examples of Fog nodes. They are usually located at the network edge (e.g., a Wi-Fi access point, a cellular base station, a camera for urban surveillance), but nodes towards the core of the network (e.g., a hardened router, a specialized server) may be exploited as well. Besides, also the patient's smartphone can be a Fog node if powerful enough, but the Fog component should be deployed to it only exceptionally (e.g., when the patient is moving out of range), in order to preserve its battery life. As Fig. 1 shows, the Fog application component is migrated to the patient's smartphone, when she gets across town. Finally, when the patient reaches the hospital, the Fog service is mi-

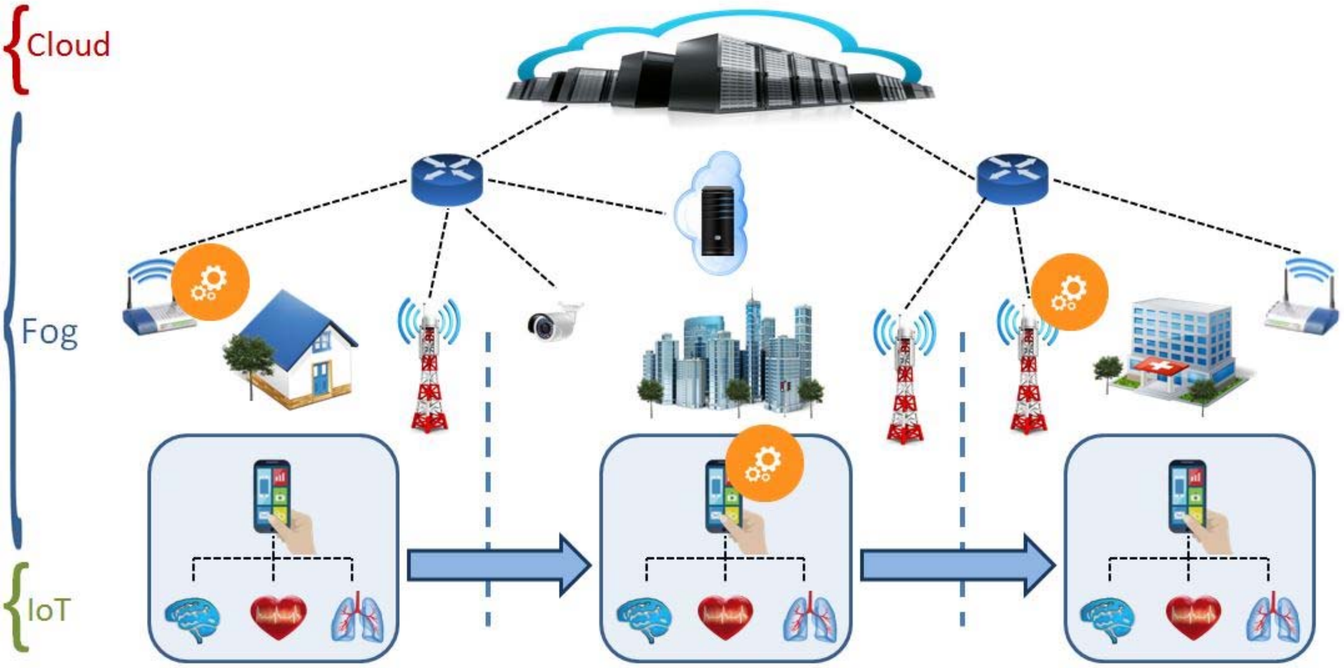

Fig. 1. The citizen's healthcare scenario.

grated to a Fog node installed in a nearby cellular base station.

2) Drones for smart urban surveillance: Drones, also known as Unmanned Aerial Vehicles (UAVs), can be employed in plenty of situations and especially in those too dangerous for humans (e.g., search for a missing person, disaster and emergency management). Indeed, drones do not have a pilot on board and are considerably cheaper than the manned aircrafts. Several companies are investing in drones, and therefore, according to [3], the global market for them is expected to reach $21.23 billion by 2022. Up to now, drones have been remotely controlled by humans, but a new generation of drones is coming. They are going to be completely autonomous, able to operate without human intervention. All processing required to analyze the collected data and make decisions may be actually performed onboard, but this would negatively affect the drone battery life and thus the overall duration of the flight. These computations are extremely resource intensive indeed, as the collected data to be analyzed are usually video streams and other sensors data, while the actions to be performed are the drone and camera control but sometimes also the action of grabbing an object. Fog Computing is the ideal paradigm to overcome this issue. Nonetheless, mobility support must be provided to the flying drone in order to make this scenario concrete. The Fog proximity to the drone is particularly needed in order to allow for a low and predictable latency and to save bandwidth. As in the previous scenario, also the drone itself may be considered as a Fog node but only in exceptional circumstances (e.g., when the drone is flying out of range), in order to preserve its battery life.

3) Tourists as time travelers: This scenario is adapted from [4]. Have you ever thought of how majestic and breathtaking could be, for example, Rome in the imperial period ? Tourists may travel back in time by taking advantage of the IoT and Fog Computing. Let us suppose that a tourist goes to Rome and takes a sightseeing bus. She brings with her a pair of interactive glasses that perceives the surrounding environment, filming whatever the wearer is looking at. The glasses stream the video to an associated application component in the Fog. This one performs very resource-intensive computations, recognizes what the tourist is currently looking at, and returns to her a visual reconstruction, to be projected on the glasses, of how that specific place was in the past. Therefore, the tourist is immersed in a virtual world where she feels to be in the right place but probably in the wrong time. It is essential that the Fog application component be continuously close enough to the tourist, following her as she moves: the need for high responsiveness and bandwidth efficiency is evident, indeed.

III. PROBLEM DESCRIPTION

As we introduced in Section I, to provide mobility support in a Fog-IoT environment means to preserve the Fog Computing advantages also when the mobile IoT nodes move away from the associated Fog application component. Since the aforementioned advantages are all made possible by the Fog proximity to the end devices, what has to be done is to migrate the Fog application component from one Fog node to another, keeping it close enough to the related mobile IoT nodes. Actually, in the event that the Fog service is stateless, there is no need to migrate it from one Fog node to another, as there is no state to be maintained. Thus, the Fog service can be simply re-instantiated on the target Fog node, and the related mobile nodes be redirected to it.

Another clarification is essential. By proximity, we actually mean topological rather than geographical: it is the communication path between the IoT nodes and the Fog application component which has to be kept short enough, indeed. Physical proximity does not always imply the logical one. Moreover, little or even no physical movements of the end devices may lead to a change of the base station and thus potentially to a considerable change in the topological distance between them and the associated Fog application component.

At a first glance, keeping the Fog component topologically close to the related IoT nodes might not seem such a hard job. Nonetheless, the problem is much more complex than what it appears to be. Many different factors, sometimes conflicting with one another, have to be taken into account, indeed. They may vary and may be given different importance from case to case, as they depend on the users' and applications requirements and on the provider's policies. Some of these factors may be: (i) Quality of Service/Quality of Experience (QoS/QoE); (ii) service continuity; (iii) security; (iv) bandwidth efficiency; and (v) load balancing.

Actually, the Fog service migration problem can be defined as the issue of determining When, Where, and How (W2H) to migrate the service in order to achieve the desired compromise among all the factors under consideration. With regard to when to migrate, one could propose to continuously migrate the Fog application component, always deploying it to the best possible Fog node, i.e., the topologically closest to where the IoT nodes are currently. However, it is not convenient to migrate the Fog service too frequently, as each migration may be very CPU- and bandwidth-consuming and implicates a minimum of downtime, i.e., a time during which the Fog service is not available.

As for where to migrate, it would be ideal to deploy the Fog component to the Fog node which is topologically closest to the mobile IoT devices. Though, the migration cost towards that node and its current workload might be taken into consideration.

Last but not least, to determine how to migrate means to decide upon which techniques and technologies to use in order to perform the service migration. Each technique may encourage certain factors more than others. Examples of these techniques are: (i) the virtualization and migration techniques; (ii) the technique exploited to identify the set of possible target Fog nodes; and (iii) the technique needed to deal with the change of the IP addresses of both the mobile nodes and the Fog service as they respectively move/migrate. This technique is required to allow for a seamless migration of the active session.

IV. RELATED WORK

The issue of mobility support in a Fog environment is drawing the attention of the research community more and more. This is mainly due to the concrete opportunities that might be unlocked by a proper handling of this thorny problem, but it is probably also a consequence of the challenging and fascinating nature that characterizes the issue itself. Moreover, the research opportunities in this field are many. The vast majority of the works that we discuss in this section is not specifically related to mobile IoT in a Fog environment, as it considers mobile nodes in general. Nonetheless, the challenges involved are very similar as well as the techniques which can be exploited to solve them. Therefore, we strongly believe that these works have to be mentioned.

Taleb et al. are very active in this field. Indeed, they first proposed the concept of Follow Me Cloud (FMC) [5] and recently advanced that of Follow Me Edge (FME) [4], which resembles the former but still distinguishes itself. Each of the two works proposes an architecture integrating different modules whose task is to manage the nodes mobility and thus the relative service migration. The FMC and FME concepts have been initially designed to support user's mobility only in cellular networks: they exploit specific elements and functionalities available in 3G, 4G, or 5G networks, indeed. With the aim to make their proposal more attractive, the authors improve the FMC concept in [6], adapting it to support mobile users connected also from networks other than the cellular one (e.g. Wi-Fi). Furthermore, in that work, they are concerned about the threat to service continuity which is raised by the change of the IP addresses after the nodes relocations. Therefore, the FMC concept is implemented exploiting the Locator/ID Separation Protocol (LISP), and all the LISP elements are virtualized according to the Network Functions Virtualization (NFV) principle.

Markov Decision Process (MDP) is a commonly used framework to formulate the service migration problem and thus to make migration decisions (i.e., when and where to migrate). In [7], the authors of FMC propose to model the service migration procedure as a distance-based MDP, i.e., an MDP where each state is the distance in terms of Service Areas between the mobile user and the associated Fog service. By doing this, the state space is considerably reduced, and the problem is more tractable. In this work, the user mobility, which is not deterministic, is modeled and predicted through a one dimension (1D) mobility pattern. In [8], Wang et al. propose their formulation of the service migration problem as a distance-based MDP. They consider the user to move according to a two dimension (2D) random walk mobility model, which is more realistic than the 1D. Assuming this mobility pattern, the authors are able to find a near-optimal service migration policy. To achieve this objective, they devise a new algorithm which distinguishes itself from the standard ones (e.g., value iteration, policy iteration) exploited in [7]. Indeed, it reduces the complexity from $O(N^3)$ to $O(N^2)$ by policy iteration, where $N+1$ is the cardinality of the state space. In [9], the authors present their proposed system, namely SEGUE. Unlike the aforementioned works, SEGUE considers also the network and Fog nodes states as parameters on which to base the migration decisions; therefore, it proves to be even more realistic. This system is composed of four modules cooperating with one another. Two of them actually collect several key parameters, analyze them, and predict if there is a possible QoS violation. If that is the case, the third module runs the MDP to determine the optimal node where to migrate the Fog service. Finally, the fourth module actually performs the migration.

The same authors of [8] advance an alternative solution method in [10]. They actually establish a decoupling property of their initial MDP which transforms it into two independent MDPs on disjoint state spaces. Lyapunov optimization can then be applied so that what is obtained is a simple deterministic (rather than stochastic) optimization problem that may be dealt with efficiently, without any knowledge of the underlying probability distributions. Another work from the same authors is the one described in [11]. It is extremely interesting, as it contextualizes the mobility support issue in military environments rather than in commercial ones. Since these environments demand strong security guarantees, a new parameter (i.e., the security cost) is considered to make a migration decision, together with the already regarded ones (i.e., transmission and migration costs). The security cost of a migration increases when services of different users would end up to be hosted on the same physical node. The problem is modeled as an MDP where there are two users, each moving according to a 1D mobility pattern, accessing two distinct services. A modified version of the myopic algorithm is proposed as a low-complexity way to solve the problem. The last but not least research work that formulates the mobility support issue as an MDP is [12]. It is worth mentioning it because the authors propose to handle user's mobility by either migrating (or re-instantiating) the application component or by finding a new, more suitable communication path between the Fog service and the mobile node. To be precise, the suggested solution consists of two correlated MDPs, each modeling either the migration decision or the new path finding one.

Although MDPs are by far the most common way to express and resolve the mobility support problem, they are not the only one. For example, in [13], the authors' objective is to make migration decisions in order to minimize the overall bandwidth consumption in a Fog-enabled Vehicular Cloud Computing (VCC) context. Here, the problem is formulated as a Mixed-Integer Quadratic Programming (MIQP) problem. A polynomial time two-phase heuristic algorithm is proposed to tackle the computational complexity which characterizes the so-formulated issue.

Few are the works specifically investigating the issue of mobility support in a Fog-IoT environment. Reference [14] conceives a hierarchical architecture where a proxy Virtual Machine (VM) running several applications in the Fog is associated to the set of the IoT devices of a user. Whenever a subset of those devices moves together with the user, only the portion of the overall proxy VM relative to those devices is migrated, following the user. The migration decisions are basically made in order to minimize the bandwidth consumption, and, to this purpose, the migration profit is defined as the difference between the total traffic amounts generated in the network for not migrating and migrating. Finally yet importantly, [15] discusses Foglets, a programming model for the geo-distributed computational continuum which goes from the sensor nodes to the Cloud, passing through the Fog. Among the other features, the authors present the APIs and algorithms proposed to handle the initial deployment and the possible migration of a Fog service. The When-to-Migrate decision is basically made considering latency as the key parameter; the Where-to-Migrate one is based on the proximity of the target Fog node to the mobile IoT devices and on the current computational workload at the target node.

V. FUTURE RESEARCH DIRECTIONS

As we have pointed out, to support mobility in a Fog/Edge Computing environment involves a considerable number of challenges: this means that there are many research opportunities and unexplored possibilities in this attractive field. In this section, we discuss the most relevant ones.

1) Proactive vs. reactive service migration: Recapping, the overall objective is to preserve the Fog benefits also when the IoT nodes move away from the associated Fog service. To this purpose, the latter has to be kept topologically close to the moving devices. That said, the horizontal or vertical handover (i.e., the transition from one base station to another) of the mobile nodes is the actual event able to cause a substantial change in the topological distance. Therefore, migration decisions (i.e., when and where) should be made, proactively or reactively, considering this event as the cornerstone. Making those decisions proactively means to make them before the actual handover, based on the stochastically predicted user's mobility. This approach is a double-edged sword. Indeed, the overall performance may benefit from it, since the Fog service is migrated beforehand and thus ready to run on the target node as soon as needed. However, as the process is stochastic, the precision in vertical handovers and user's mobility predictions may be low. This can result in a possible risk to overall performance. On the other hand, a reactive approach makes service migration decisions only after a handover has happened. By doing so, there is determinism, and there is no real difference between vertical and horizontal handovers. Nonetheless, in this case, the Fog service is not migrated beforehand. Combined approaches should also be investigated in order to get the most out of the two.

2) Exploit context information to trigger service migration: Another interesting question is that of determining on which parameters to base the migration decisions. These parameters vary from case to case, as they depend on the actual objectives (e.g., QoS, bandwidth efficiency) among which to find the desired compromise. Previous work just considered system and network parameters, such as: (i) network state; (ii) Fog nodes state; (iii) amount of data exchanged by the application; and (iv) migration costs in terms of bandwidth consumption and time. However, it might be useful to take advantage of the context awareness enabled by the Fog: thus, context information may be considered as a possible parameter, together with the network and system ones. For example, with regard to the patient's scenario in Section II, the information that the patient is within the area of the hospital can be exploited by the service migration policy to also consider the Fog nodes belonging to the hospital among the eligible target nodes.

3) Fog federation to enable mobile roaming: It might be interesting to conceive mobility support solutions that exploit a federation among Fog domains. Indeed, mobility support may be dramatically improved if considering a federated Fog environment. For example, let us suppose that a user moves to an area in which there are not Fog nodes belonging to her Fog domain. In case of federation, the user could rely on a Fog node associated to a federated Fog domain (i.e., higher availability). Similarly, let us assume that a user moves to an area in which there exists a Fog node belonging to her domain, but this is topologically too distant for her needs. Federation could do the trick by providing the user with a considerably closer Fog node that belongs to a federated domain (i.e., better performance). In order to realize a federation among Fog domains, there are several challenges to be faced, the majority of which has not been dealt with yet. Some of these challenges are: (i) the management of the Service Level Agreements (SLAs) among Fog domains; (ii) which architecture to use, centralized or distributed peer-to-peer; and (iii) the actual technologies to exploit.

4) Virtualization and migration techniques: Carefully choosing the virtualization and migration techniques can substantially improve a mobility support solution both in terms of performance and applicability. For example, it could be desirable to find a virtualization technology suiting as many different types of physical nodes as possible, thus to expand the set of nodes qualified to host a Fog service. Equivalently, it is of vital importance to identify a migration technique able to minimize the overall duration of the migration and of the experienced downtime as well as the bandwidth consumption. Some of these techniques are proposed in [16] and [17], but there is still room for further investigation. Finally, there already exist some techniques (e.g., Mobile IP and Mobile IPv6, LISP, Distributed Mobility Management) for handling the change of the IP addresses as nodes move, thus to ensure a seamless migration of the active session. Nonetheless, there is still the possibility to conceive new techniques or improve the existing ones.

5) Compliance with existing interoperability platforms: When coming up with a new mobility support solution, one approach could be that of defining also the reference architecture and the APIs to be used. However, it may be convenient and effective to exploit an already existing standard and/or open platform (e.g., oneM2M, FIWARE, AllJoyn, IoTivity), with its specified architecture and APIs, and extend its features to include those belonging to the proposed solution. By doing this, researchers would allow for a non-negligible improvement in the considered platform, and, at the same time, they would earn visibility and prestige.

6) Integration with mobile networks towards 5G: Mobile Edge Computing (MEC) is a standard Fog Computing system defined by the European Telecommunications Standards Institute (ETSI) [18]. It is considered as one of the key enablers of IoT services and of the forthcoming 5G networks. An interesting research direction could be that of proposing a mobility support solution as part of the ETSI MEC system. Thereby, researchers could leverage the system itself but also the set of legacy core network functionalities which already deal with some of the mobility and session management issues. For example, the Mobility Management Entity (MME) [19] component in the Long Term Evolution (LTE) core network provides some useful functionalities such as: (i) retention of users' location information; (ii) selection of the appropriate gateway during the initial registration process; (iii) management of the handover between LTE and 2G/3G networks; (iv) management of the roaming from other LTE or legacy networks; and (v) session management. All these functionalities could play an important role in the resolution of the discussed mobility support problem; however, how to best exploit them is still an open research issue.

VI. CONCLUSIONS

Fog Computing was proposed in 2012 as the ideal paradigm to support the resource-constrained IoT devices in data processing and information delivery. Its topological proximity to those devices is the key enabler of several advantages that were not possible when continuously offloading onto the distant Cloud. Nodes mobility is a threat to the preservation of these advantages, as, when nodes move, the topological distance between them and the associated Fog service may be seriously compromised. Therefore, the latter needs to be migrated in order to follow the related mobile IoT nodes and guarantee the topological proximity to them.

In this paper, we have pointed out the relevance of this topic in everyday life by presenting three scenarios where mobility support is essential. Moreover, we have investigated in depth what it means to provide mobility support in a Fog-IoT environment, the related work in the field, and the main research directions.

ACKNOWLEDGMENT

This work has been partially supported by the project "IoT and Big Data: methodologies and technologies for large scale data gathering and processing" - PRA_2017_37, funded by University of Pisa.

REFERENCES

[1] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog computing and its role in the Internet of Things”, Mobile Cloud Computing (MCC) Conference, August 17, 2012, Helsinki, Finland. doi: 10.1145/2342509.2342513

[2] "Cisco Visual Networking Index: global mobile data traffic forecast update, 2016–2021", Cisco White paper, February 7, 2017.

[3] MarketsandMarkets, "Drones market worth 21.23 Billion USD by 2022", 2016. [Online]. Available: http://www.marketsandmarkets.com /PressReleases/commercial-drones.asp. [Accessed: 13- Mar- 2017].

[4] T. Taleb, S. Dutta, A. Ksentini, M. Iqbal, and H. Flinck, "Mobile Edge Computing potential in making cities smarter", IEEE Communications Magazine, vol. 55, issue 3, pp. 38-43, March 2017. doi: 10.1109/MCOM.2017.1600249CM

[5] T. Taleb, A. Ksentini, "Follow Me Cloud: interworking federated clouds and distributed mobile networks", IEEE Network, vol. 27, issue 5, pp. 12-19, October 2013. doi: 10.1109/MNET.2013.6616110

[6] A. Ksentini, T. Taleb, and F. Messaoudi, "A LISP-based implementation of Follow Me Cloud", IEEE Access, vol. 2, pp. 1340-1347, September 2014. doi: 10.1109/ACCESS.2014.2360352

[7] A. Ksentini, T. Taleb, and M. Chen, "A Markov Decision Process-based service migration procedure for Follow Me Cloud", IEEE International Conference on Communications (ICC), June 10-14, 2014, Sidney, Australia. doi: 10.1109/ICC.2014.6883509

[8] S. Wang, R. Urgaonkar, M. Zafer, T. He, K. Chan, and K. K. Leung, "Dynamic service migration in mobile edge-clouds", IFIP Networking Conference, May 20-22, 2015, Toulouse, France. doi: 10.1109/IFIPNetworking.2015.7145316

[9] W. Zhang, Y. Hu, Y. Zhang, and D. Raychaudhuri, "SEGUE: Quality of Service aware edge cloud service migration", IEEE International Conference on Cloud Computing Technology and Science (CloudCom), December 12-15, 2016, Luxembourg City, Luxembourg. doi: 10.1109/CloudCom.2016.0061

[10] R. Urgaonkar, S. Wang, T. He, M. Zafer, K. Chan, and K. K. Leung, "Dynamic service migration and workload scheduling in edge-clouds", Performance Evaluation, vol. 91, pp. 205-228, September 2015. doi: 10.1016/j.peva.2015.06.013

[11] E. N. Ciftcioglu, K. S. Chan, R. Urgaonkar, S. Wang, and T. He, "Security-aware service migration for tactical mobile micro-clouds", IEEE Military Communications Conference (MILCOM), October 26-28, 2015, Tampa, FL, USA. doi: 10.1109/MILCOM.2015.7357585

[12] J. Plachy, Z. Becvar, and E. C. Strinati, "Dynamic resource allocation exploiting mobility prediction in Mobile Edge Computing", IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), September 4-8, 2016, Valencia, Spain. doi: 10.1109/PIMRC.2016.7794955

[13] H. Yao, C. Bai, D. Zeng, Q. Liang, and Y. Fan, "Migrate or not ? Exploring virtual machine migration in roadside cloudlet-based vehicular cloud", Concurrency and Computation: Practice and Experience, vol. 27, issue 18, pp. 5780-5792, December 2015. doi: 10.1002/cpe.3642

[14] X. Sun, and N. Ansari, "EdgeIoT: Mobile Edge Computing for the Internet of Things", IEEE Communications Magazine, vol. 54, issue 12, pp. 22-29, December 2016. doi: 10.1109/MCOM.2016.1600492CM

[15] E. Saurez, K. Hong, D. Lillethun, U. Ramachandran, and B. Ottenwalder, "Incremental deployment and migration of geo-distributed situation awareness applications in the Fog", DEBS '16 Proceedings of the 10th ACM International Conference on Distributed and Event-based Systems, pp. 258-269, June 20-24. 2016, Irvine, CA, USA. doi: 10.1145/2933267.2933317

[16] K. Ha, Y. Abe, Z. Chen, W. Hu, B. Amos, P. Pillai, and M. Satyanarayanan, "Adaptive VM handoff across Cloudlets", Technical Report CMU-CS-15-113, CMU School of Computer Science, June 2015.

[17] A. Machen, S. Wang, K. K. Leung, B. J. Ko, and T. Salonidis, "Migrating running applications across mobile edge clouds", MobiCom '16 Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking, pp. 435-436, October 3-7, 2016, New York, NY, USA. doi: 10.1145/2973750.2985265

[18] Y. Chao Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile Edge Computing - a key technology towards 5G", ETSI White paper, September 2015.

[19] Alcatel-Lucent, "Introduction to Evolved Packet Core", Strategic White paper, 2009.