Accepted at: Workshop on Fog Computing and the IoT (IoT-Fog), pp. 1–5, 2019, ACM

Edge Computing with Peer to Peer Interactions: Use Cases and Impact

Vasileios Karagiannis, Michael Borkowski
Distributed Systems Group, TU Wien, Austria
{v.karagiannis, m.borkowski}@infosys.tuwien.ac.at

Alexandre Venito, Rodrigo Coelho, Gerhard Fohler
Chair of Real-Time Systems, TU Kaiserslautern, Germany
{venito, coelho, fohler}@eit.uni-kl.de

***Abstract*—Edge and fog computing applications typically focus on outsourcing computations from static or mobile end user devices towards compute nodes in proximity. However, such applications do not fully exploit the benefits of the edge because the resources of neighboring end user devices are not considered as potential compute nodes. For this reason, we propose complementing edge computing with peer to peer interactions in order to enable the end user devices to communicate with each other and share computations. Peer to peer is a well established communication model that can be used for organizing the available resources based on proximity disregarding their role in the network (i.e., end user device, edge or cloud node). This way, all resources become accessible and computations may be outsourced towards any node. In this paper, we present three edge computing use cases that can benefit from the use of peer to peer interactions. To further motivate the use of peer to peer, we analyze the compatibility with edge computing and the potential impact, and we identify related research directions.**

***Index Terms*—Edge computing, Fog computing, Internet of Things, Peer to Peer, Distributed Applications, Distributed Architectures, Use Cases**

I. Introduction

When the term edge computing was coined, it was used mainly to refer to distributed applications hosted on content delivery networks so that the end users could benefit from being in the proximity of the servers [1]. However, with the advent of the Internet of Things (IoT), the edge gained further popularity as a means to host applications on IoT-inherent compute resources [2]. The IoT envisions a plethora of devices that communicate with each other to facilitate applications with sensing, actuating and processing features [3]. IoT applications pledge to improve the general well being of individuals and societies by increasing the use of technology in health care, transportation, manufacturing etc. [4]. However, this approach generates an immense amount of data that may cause bottlenecks on the underlying network [5]. Edge computing emerged to deal with this problem by pushing the computations from the cloud towards the edge, which aims at avoiding bottlenecks and reducing latency [6].

Notable advances in the field of edge computing focus on end user applications able to outsource computations on compute nodes at the edge of the network–also referred to as *fog nodes* [7]. For instance, the ETSI Mobile Edge Computing specification describes an architecture for deploying compute nodes on base stations in order to serve mobile users [8]. Moreover in [9], the authors study the benefits that stem from outsourcing computations from end user devices to compute nodes in proximity. Furthermore, [10] shows that for small computations (e.g., from IoT devices) the processing delay of outsourcing to fog nodes is always lower than outsourcing to the cloud.

Even though current trends enable outsourcing computations to compute nodes in proximity, the resources of neighboring end user devices are not considered as potential compute nodes. This hampers the exploitation of the available resources at the edge of the network. Therefore, in contrast to the aforementioned studies, this paper considers all involved resources as compute nodes that may request to either outsource computations or offer compute resources to neighboring nodes. Moreover, in this paper we motivate the use of the peer to peer (P2P) communication model for interconnecting these compute nodes. Contrarily to the typical communication in edge computing [11], nodes in P2P networks are not organized hierarchically [12]. Instead, P2P nodes follow a flat model whereby each node maintains a limited number of neighbors based on proximity [13]. For this reason, we propose complementing edge computing with P2P in order to utilize these neighbors for outsourcing computations so that all the available resources at the edge of the network are exploited.

P2P is a well established communication model that can aid in realizing computing at the edge because P2P systems allow endpoints to cooperate with each other in order to achieve common goals [14]. In addition, P2P can act as an orchestration model that schedules and coordinates applications for meeting Quality of Service (QoS) requirements [15]. Moreover, P2P has shown potential for handling distributed infrastructures in a scalable manner [16]. Due to this scalability, P2P fits scenarios with a large number of devices, such as the IoT. However, the IoT also includes resource constrained nodes that cannot implement complex communication mechanisms [17]. Nevertheless, such nodes can still join in P2P applications by connecting to a compute node that acts as a gateway to/from the P2P network. The P2P model can also benefit from edge computing since stable resources at the edge can aid in resolving fault tolerance and transient availability issues of P2P systems [1].

Therefore, the aim of the work at hand is to motivate P2P

This paper has received funding from the H2020 FORA—Fog Computing for Robotics and Industrial Automation—under the grant agreement No. 764785.

interactions at the edge by describing three edge computing use cases that benefit from P2P. Namely, these use cases are related to: *i*) suspect identification/finding lost children (Section II-A), *ii*) augmented reality applied in tourism and sightseeing (Section II-B) and *iii*) warehouse communication for online shopping (Section II-C). We further motivate the use of P2P by analyzing the compatibility with edge computing (Section III-A) and the potential impact (Section III-B).

II. Use Cases

This section analyzes edge computing use cases in which P2P interactions introduce additional functionality. For each use case we present, we describe why this use case belongs to the edge computing field, how an application that addresses this use case operates and how P2P interactions improve this application.

A. Suspect Identification/Finding Lost Children

The first use case, suspect identification/finding lost children, refers to identifying a person of interest using video feed from multiple smart cameras [18]. The smart cameras (i.e., cameras attached to a small sized compute node) may be scattered over a neighborhood, attached to police vehicles or to unarmed flying drones. We consider this as an edge computing use case because *i*) the processing can be distributed among the nodes at the edge (i.e., the devices that host the cameras) and *ii*) because when identifying a person of interest, there is usually an important clue which is the last known location that the person was seen. Edge computing leverages on exploiting intelligence at the edge which can be applied to this use case by starting the search of a person from the last known location.

In this use case, typically, there is a central management system that collects the video feed from the cameras and performs resource intensive processing to extract valuable information [19]. Thus, to facilitate a surveillance and identification application, a central station compares the video feed with a reference image of the person of interest for potentially matching this image to the video feed. However, at the same time the resources of the smart cameras used for capturing the video feed may be operating at low capacity while they could also be performing part of the comparing computations.

By applying P2P interactions in this use case, we assume that instead of the central station, there is a fog node as shown in Fig. 1. The fog node is part of the same P2P network as the smart cameras and dedicated compute nodes in road side units. When there is a request to search for a person of interest, the fog node sends the reference image to the compute node (i.e., smart camera) closest to the last known location. Then, this compute node actively compares the reference image with the live feed of the camera but also sends the image to smart cameras in proximity, which are the neighbors in the P2P network. Each smart camera that receives the reference image forwards it to the neighbors, which leads to a global broadcast starting from the last known location and spreading epidemically to the nodes in proximity. The compute node that matches the reference image, responds to the fog node and notifies the smart cameras (again in an epidemic manner) so that the processing stops. The smart cameras that do not have enough resources to host the comparing computations can outsource to roadside units, neighboring smart cameras or the fog node. This way, all the available resources at the edge of the network are exploited.

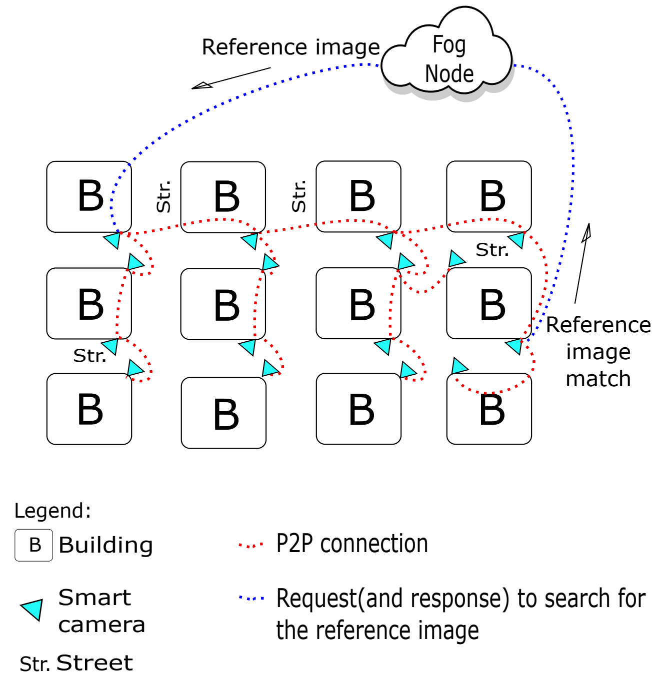

Fig. 1: Peer to peer interactions among the smart cameras.

B. Augmented Reality Applied in Tourism and Sightseeing

This use case refers to the use of augmented reality in tourism and sightseeing. Augmented reality applications simulate visualizations of historic events, features and objects, and render these visualizations into the landscape of the video feed of a person's digital camera (e.g., using smartphones or tablets) [20]. This way, tourists can access real time information about a point of interest, e.g., a historic monument, while filming the monument using smartphones. The real time information may include historic representations of events, narrative explanations and comments from the organizers or other users [21].

However, augmented reality applications require intense data processing while being very sensitive to latency delay and therefore, executing these applications on mobile devices is prohibitive due to the limited lifespan of the devices' batteries [22]. Since performing the processing in the cloud leads to high latency, augmented reality is considered a use case for edge computing [23], [24]. However, the sightseeing scenario is faced with the additional challenge of high user density. The amount of users around the location of a point of interest and the respective requests for video rendering, can overload a fog node leading to bottlenecks and high latency delays. For this reason, we propose the use of P2P interactions

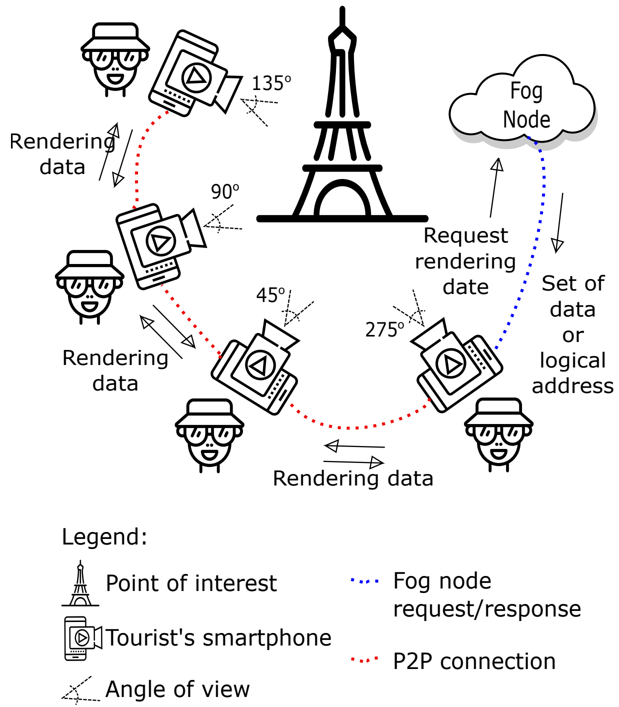

Fig. 2: Peer to peer interactions among the tourists' smartphones.

to interconnect all the compute nodes around each point of interest.

We assume that a new user approaching a point of interest requests data rendering from a fog node as shown in Fig. 2. The fog node examines the angle of view of the new user and checks if there is another device in the network with similar angle. If there is no other device with similar angle, the compute node executes the data rendering and sends the requested data to the new user. If there is another device with similar angle, the fog node responds with the logical address of the device that already has the data rendering information. Then, the new user forms a P2P connection with this device and downloads the requested data. Considering that all users around the sight are interested in similar information but with little difference in the angle of view, using P2P interactions among the visitors' devices can greatly reduce the load of the fog node.

C. Warehouse Communication for Online Shopping

Online shopping has been proposed as a use case of edge computing for reducing the latency of adding products to the client's shopping cart in order to improve the quality of service (QoS) [25]. However, apart from increasing the QoS at the front end, edge computing can further improve online shopping by leveraging on the proximity between warehouses. This can be achieved by ordering a product from the warehouse closest to the location of the client. If the product is not available there, the warehouse can forward the order to another warehouse in proximity. This leads to minimizing the delivery time.

The warehouses of an online shop may be dispersed all over the world and new ones might be added dynamically due to new buildings or business collaborations. Keeping a global snapshot of the stock in a central station is the cloud based approach which can be prone to bottlenecks due to high demand [26]. Contacting all the warehouses separately requires time and is wasteful in terms of utilization of network resources since independent inquires ignore that some warehouses might be very close to each other and forwarding may improve bandwidth utilization [27]. For this reason, we believe that this scenario can leverage on P2P interactions.

We assume that there is a fog node that handles the shopping cart of the client as described in [25]. However, we also assume that this fog node is part of the same P2P network with the compute nodes of the warehouses as shown in Fig. 3. The client submits the order along with his location to the fog node which forwards the order to the warehouse closest to the client. Depending on the products of the order the compute node of the closest warehouse checks the stock to examine if the selected products are available locally. Consequently, the available products are ordered from the closest warehouse. For unavailable products, the warehouse forwards the order (with the pending products) to the closest neighbor of the P2P network which repeats the process. This way, the order is initially processed by a warehouse near the client and after that, by the warehouses in proximity using P2P interactions. Eventually, all the products of the order are requested from the closest warehouse (based on availability).

III. Compatibility and Impact

In this section, we describe the prerequisites of P2P and comment on the compatibility with edge computing (Sec-

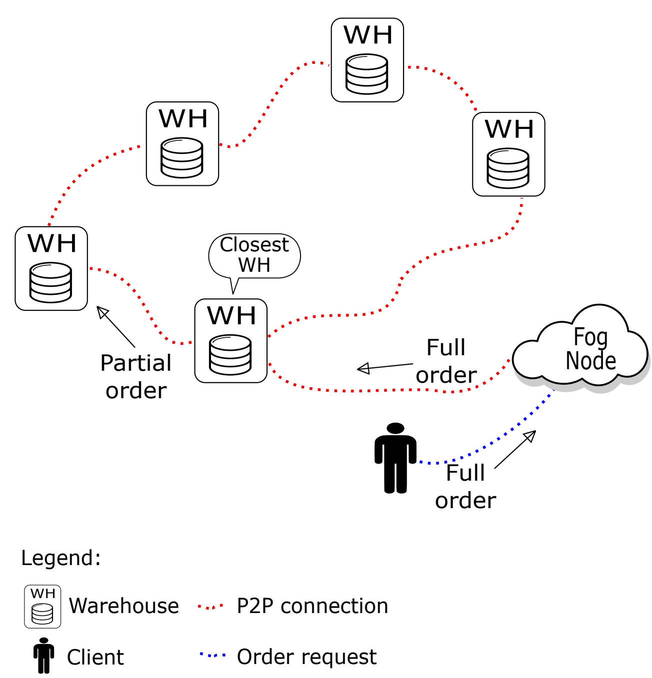

Fig. 3: Peer to peer interactions among the warehouses.

tion III-A). Then, we analyze the impact of leveraging P2P in edge computing by discussing challenges and requirements of edge computing which can be met by research conducted within the context of P2P systems (Section III-B). Finally, we identify research directions related to realizing P2P interaction for edge computing.

A. Compatibility of Peer to Peer with Edge Computing

P2P protocols operate on the application layer and assume an underlying network with IP routing capabilities and an IP-based transport layer protocol. When the goal is to transfer content in real time (e.g., live streaming or voice calls) the preferred transport layer protocol is UDP since lost packets cease being useful after short periods of time and retransmissions induce unnecessary overhead. Otherwise, P2P protocols can be implemented on top of the reliable TCP or HTTP [28]. Thus, in a P2P network, all participating nodes are assumed to have an IP address and a port number through which they can be reached by other participants. Non-compatible devices (e.g., resource constrained devices) can still join a P2P network through a node assigned with an IP address that acts as a gateway to/from the rest of the network.

Edge computing assumes an infrastructure with geographically distributed compute resources. In contrast to P2P, these resources are organized hierarchically with cloud compute nodes at the high level, compute nodes at the edge of the network in the middle and the resource constrained devices at the low level [29]. Despite the different organization, P2P and edge computing use similar underlying communication (i.e., transport layer protocols). Nevertheless, edge computing uses additional mechanisms, e.g., for provisioning virtual resources on the participating compute nodes in an automatic way [30]. For this reason, we propose complementing with P2P interactions, since we believe that edge computing can benefit from the decades of research and implementation of P2P systems.

B. Impact of Peer to Peer on Edge Computing

The prospective impact of P2P on edge computing stems from requirements of edge computing applications that have already been dealt with in the context of P2P. For this reason, we identify problems of P2P applications that have been solved in the literature, but are still likely to occur in edge computing. In the following, we present such problems and discuss why they pose challenges for edge computing applications.

- *Churn management*. Maintaining connectivity when some of the participating nodes fail or disconnect unexpectedly from the network, is called churn management. In edge computing, this is one of the factors that can severely degrade QoS and user satisfaction due to the induced latency [31]. However, churn management for P2P has been discussed in [32].
- *Scalability*. Edge computing will result in massive overlay networks which can pose a challenge for the communication protocols [1]. However, P2P is a highly scalable communication model. For instance, the authors of [33], discuss a P2P overlay network comprising more than a million nodes, which is able to provide stable and efficient connectivity despite the rapid dynamics of node participation.
- *Routing*. Routing is an essential mechanism for managing and orchestrating applications in edge computing environments [34]. Nevertheless, within the context of P2P systems, there is a lot of related literature on routing strategies [35].
- *Proximity awareness*. Edge computing neighboring nodes are close-by in a physical/logical sense [1]. However, leveraging on proximity was the main motivation of P2P systems which already implement mechanisms for proximity awareness [36]. Moreover, the same mechanisms also deal with resource discovery which is still a challenge for edge computing [37].
- *Security*. Security and privacy is an important challenge for edge computing due to the distributed network topology and the unreliable participating devices [5], [37]. However, similar security and privacy issues have been discussed in P2P systems [38].

By considering this list with challenges of edge computing and the respective literature that proposes solutions from P2P systems, we deduce that complementing edge computing with P2P interactions promises a positive impact on future edge computing applications.

However, realizing P2P interactions in edge computing is not trivial, which introduces potential research directions. So far, P2P overlay networks do not support mechanisms for provisioning virtual resources. Moreover, P2P networks do not consider mixed criticality environments which is a domain that will be addressed by fog and edge computing [39], [40]. In such environments, the applications may compete with each other for the utilization of the available resources in order to meet real time requirements [41]. Therefore, these limitations of P2P introduce potential research directions which should aim at proposing novel P2P networks specifically designed for edge computing.

IV. Conclusion

Edge computing enables outsourcing computations to compute nodes in the proximity of the data source, which leads to low latency and improved bandwidth utilization. However, there are still many challenges in the edge computing field, which are related to: churn management, scalability, routing, proximity awareness and security. In this paper, we propose the use of mechanisms from P2P systems for addressing these challenges and we identify related research directions.

To further motivate the use of P2P interactions, we present three use cases, namely: *i*) suspect identification/finding lost children, *ii*) augmented reality applied in tourism and sight-seeing and *iii*) warehouse communication for online shopping. These use cases show how typical edge computing applications can benefit from the use of P2P.

REFERENCES

[1] P. Garcia Lopez, A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, and E. Riviere, "Edge-centric computing: Vision and challenges," *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 5, pp. 37–42, 2015.

[2] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Workshop on Mobile Cloud Computing (MCC)*, pp. 13–16, ACM, 2012.

[3] V. Karagiannis, P. Chatzimisios, F. Vazquez-Gallego, and J. Alonso-Zarate, "A survey on application layer protocols for the internet of things," *ICAS Transaction on IoT and Cloud Computing*, vol. 3, no. 1, pp. 11–17, 2015.

[4] L. Hou, S. Zhao, X. Xiong, K. Zheng, P. Chatzimisios, M. S. Hossain, and W. Xiang, "Internet of things cloud: Architecture and implementation," *IEEE Communications Magazine*, vol. 54, no. 12, pp. 32–39, 2016.

[5] W. Shi and S. Dustdar, "The promise of edge computing," *IEEE Computer*, vol. 49, no. 5, pp. 78–81, 2016.

[6] V. Karagiannis and A. Papageorgiou, "Network-integrated edge computing orchestrator for application placement," in *International Conference on Network and Service Management (CNSM)*, pp. 1–5, IEEE, 2017.

[7] M. Satyanarayanan, "The emergence of edge computing," *Computer*, vol. 50, no. 1, pp. 30–39, 2017.

[8] "Mobile edge computing (mec): Framework and reference architecture," in *GS MEC 003*, ETSI, 2016.

[9] W. Hu, Y. Gao, K. Ha, J. Wang, B. Amos, Z. Chen, P. Pillai, and M. Satyanarayanan, "Quantifying the impact of edge computing on mobile applications," in *Asia-Pacific Workshop on Systems (SIGOPS)*, p. 5, ACM, 2016.

[10] M. Aazam, S. Zeadally, and K. A. Harras, "Fog computing architecture, evaluation, and future research directions," *IEEE Communications Magazine*, vol. 56, no. 5, pp. 46–52, 2018.

[11] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick, and D. S. Nikolopoulos, "Challenges and opportunities in edge computing," *arXiv preprint arXiv:1609.01967*, 2016.

[12] V. Karagiannis, "Compute node communication in the fog: Survey and research challenges," in *Workshop on Fog Computing and the IoT (IoT-Fog '19)*, pp. 1–5, ACM, 2019.

[13] G. Tato, M. Bertier, and C. Tedeschi, "Designing overlay networks for decentralized clouds," in *Cloud Computing Technology and Science (CloudCom), 2017 IEEE International Conference on*, pp. 391–396, IEEE, 2017.

[14] L. M. Vaquero and L. Rodero-Merino, "Finding your way in the fog: Towards a comprehensive definition of fog computing," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 27–32, 2014.

[15] P. Varshney and Y. Simmhan, "Demystifying fog computing: Characterizing architectures, applications and abstractions," in *International Conference on Fog and Edge Computing (ICFEC)*, pp. 115–124, IEEE, 2017.

[16] A. Lebre, J. Pastor, A. Simonet, and F. Desprez, "Revising openstack to operate fog/edge computing infrastructures," in *Cloud Engineering (IC2E), 2017 IEEE International Conference on*, pp. 138–148, IEEE, 2017.

[17] O. Bello, S. Zeadally, and M. Badra, "Network layer inter-operation of device-to-device communication technologies in internet of things (iot)," *Ad Hoc Networks*, vol. 57, pp. 52–62, 2017.

[18] A. Sankaranarayanan, A. Veeraraghavan, and R. Chellappa, "Object Detection, Tracking and Recognition for Multiple Smart Cameras," *Proceedings of the IEEE*, vol. 96, pp. 1606–1624, Oct. 2008.

[19] D. Radu, A. Cretu, C. Avram, A. Astilean, and B. Parrein, "Video content transmission in a public safety system model based on flying ad-hoc networks," in *2018 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, pp. 1–4, IEEE, 2018.

[20] C. D. Kounavis, A. E. Kasimati, and E. D. Zamani, "Enhancing the tourism experience through mobile augmented reality: Challenges and prospects," *International Journal of Engineering Business Management*, vol. 4, p. 10, 2012.

[21] P. J. Bartie and W. A. Mackaness, "Development of a Speech-Based Augmented Reality System to Support Exploration of Cityscape," *Transactions in GIS*, vol. 10, pp. 63–86, Jan. 2006.

[22] A. Al-Shuwaili and O. Simeone, "Energy-efficient resource allocation for mobile edge computing-based augmented reality applications," *IEEE Wireless Communications Letters*, vol. 6, no. 3, pp. 398–401, 2017.

[23] 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, 2015.

[24] A. V. Dastjerdi and R. Buyya, "Fog computing: Helping the internet of things realize its potential," *Computer*, vol. 49, no. 8, pp. 112–116, 2016.

[25] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[26] C. Sonmez, A. Ozgovde, and C. Ersoy, "Performance evaluation of single-tier and two-tier cloudlet assisted applications," in *International Conference on Communications Workshops (ICC Workshops)*, pp. 302–307, IEEE, 2017.

[27] I. Rodero, F. Guim, J. Corbalan, L. Fong, and S. M. Sadjadi, "Grid broker selection strategies using aggregated resource information," *Future Generation Computer Systems*, vol. 26, no. 1, pp. 72–86, 2010.

[28] M. Janbeglou and N. Brownlee, "Overudp: Tunneling transport layer protocols in udp for p2p application of ipv4," in *International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, pp. 325–330, IEEE, 2016.

[29] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, and X. Yang, "A survey on the edge computing for the internet of things," *IEEE access*, vol. 6, pp. 6900–6919, 2018.

[30] F. Bonomi, R. Milito, P. Natarajan, and J. Zhu, "Fog computing: A platform for internet of things and analytics," in *Big data and internet of things: A roadmap for smart environments*, pp. 169–186, Springer, 2014.

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

[32] D. Stutzbach and R. Rejaie, "Understanding churn in peer-to-peer networks," in *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, pp. 189–202, ACM, 2006.

[33] D. Stutzbach, R. Rejaie, and S. Sen, "Characterizing unstructured overlay topologies in modern p2p file-sharing systems," *IEEE Transactions on Networking (ToN)*, vol. 16, no. 2, pp. 267–280, 2008.

[34] D. Chemodanov, F. Esposito, A. Sukhov, P. Calyam, H. Trinh, and Z. Oraibi, "Agra: Ai-augmented geographic routing approach for iot-based incident-supporting applications," *Future Generation Computer Systems*, vol. 92, pp. 1051–1065, 2019.

[35] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim, "A survey and comparison of peer-to-peer overlay network schemes," *IEEE Communications Surveys & Tutorials*, vol. 7, no. 2, pp. 72–93, 2005.

[36] M. Castro, P. Druschel, Y. C. Hu, and A. Rowstron, "Topology-aware routing in structured peer-to-peer overlay networks," in *Future directions in distributed computing*, pp. 103–107, Springer, 2003.

[37] P. Hu, S. Dhelim, H. Ning, and T. Qiu, "Survey on fog computing: architecture, key technologies, applications and open issues," *Journal of Network and Computer Applications*, vol. 98, pp. 27–42, 2017.

[38] G. Gheorghe, R. L. Cigno, and A. Montresor, "Security and privacy issues in p2p streaming systems: A survey," *Peer-to-Peer Networking and Applications*, vol. 4, no. 2, pp. 75–91, 2011.

[39] K. S. Desikan, V. J. Kotagi, and C. S. R. Murthy, "Smart at right price: A cost efficient topology construction for fog computing enabled iot networks in smart cities," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–7, IEEE, 2018.

[40] A. Roy, C. Roy, S. Misra, Y. Rahulamathavan, and M. Rajarajan, "Care: criticality-aware data transmission in cps-based healthcare systems," in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, IEEE, 2018.

[41] G. Fohler, G. Gala, D. Gracia Pérez, and C. Pagetti, "Evaluation of dreams resource management solutions on a mixed-critical demonstrator," in *9th European Congress on Embedded Real Time Software and Systems (ERTS)*, 2018.