Towards Service Protection in Fog-to-Cloud (F2C) Computing Systems

Vitor Barbosa Souza[§*], Wilson Ramírez[*], Xavi Masip-Bruin[*], Eva Marín-Tordera[*], Sergio Sánchez-López[*], Guang-Jie Ren[‡]

§ Informatics Department (DPI), Universidade Federal de Viçosa (UFV), Brazil
*Advanced Network Architectures Lab (CRAAX), Universitat Politècnica de Catalunya (UPC), Spain
‡ IBM Almaden Research Center, USA
Email: vitorbs@dpi.ufv.br, {wramirez, xmasip, eva, sergio}@ac.upc.edu, gren@us.ibm.com

***Abstract*—Internet of Things (IoT) services are unstoppably demanding more computing and storage resources. Aligned to this trend, cloud and fog computing came up as the proper paradigms meeting such IoT services demands. More recently, a new paradigm, so-called fog to cloud (F2C) computing, promises to make the most out of both Fog and Cloud, paving the way to new IoT services development. Nevertheless, the benefits of F2C architectures may be diminished by failures affecting the computing commodities. In order to withstand possible failures, the design of novel protection strategies, specifically designed for distributed computing scenarios is required. In this paper, we study the impact of distinct protection strategies on several key performance aspects, including service response time, and usage of computing resources. Numerical results indicate that under distinct failure scenarios, F2C significantly outperforms the conventional cloud.**

Keywords—Cloud computing; fog computing; fog-to-cloud computing; Internet of Things; service protection

I. INTRODUCTION

In recent years, Cloud Computing has gained momentum in future Internet of Things (IoT) scenarios, such as Smart Cities or Smart Transportation. This is mainly because Cloud computing properly addresses the ever increasing requirements of IoT services, related to both computing and storage capabilities [1]. Nevertheless, Cloud computing faces substantial challenges, such as large response time and global mobility support. These challenges cannot be overlooked for the envisioned massive IoT deployment.

Fortunately, the advent of a new computing paradigm referred to as Fog computing promises to overcome the negative issues linked to Cloud [2]. The rationale behind Fog Computing is to move computing resources to the edge of the network, bringing two key benefits. First, IoT services can be deployed closer to the end-user, resulting in lower service response time. Second, the network core lightens its load by reducing traffic to/from cloud. Despite the fact that Fog computing cannot provide the massive storage capacity at cloud, Fog is more suitable for services requiring real time processing such as Healthcare or Smart Transportation. Indeed, rather than competing with Cloud, Fog computing looks forward to a scenario where both Cloud and Fog architectures collaborate to ease IoT services deployment.

Motivated by the potential benefits to come from bringing together Fog and Cloud Computing, a new network architecture referred to as Fog-to-Cloud (F2C) computing has been recently proposed [3]. F2C leverages a hierarchical organization of existing resources into different layers, aiming at easing the development of new services, not properly supported by current fog or cloud computing paradigms, such as those based on collaborative models. However, albeit several solutions for failure recovery have been proposed and successfully deployed in data center networks, resilience in fog computing is still an open challenge [4]. These concerns are inherited by F2C computing systems raising several challenges regarding augmented disruption probability caused by the high dynamicity observed in this architecture.

In this work, we consider for service protection both the failure recovery delay and Protection Cost (P_{cost}), which is measured in terms of amount of resources reserved for recovery of eventual failures. This work is an extension of the one presented in [5], where failure recovery in F2C systems is assessed for the first time. However, it is with no doubt that more research efforts must be devoted to distill the challenges related to service protection in F2C scenarios. To fill this gap, this paper introduces and formulates the so-called Protected Service Allocation (PSA) problem in F2C scenarios. Put simply, the main goal of the PSA problem is to minimize the service response time and the P_{cost} considering both the amount of protection resources and their respective layer in the F2C topology, while avoiding the commodity nodes to get overloaded.

The rest of this paper is organized as follows. Section II discusses previous works dealing with fail recovery. Section III introduces the architectural model of the F2C architecture as well as the PSA problem formulation. In Section IV, simulation results are presented and discussed while Section V concludes the paper and suggests avenues for future work.

II. RELATED WORK

In this section, we highlight the main recent research studies dealing with the resilience of both Cloud and Fog Computing. There are myriad of research studies dealing with several resilience aspects of Cloud and Fog Computing. On one hand, studies available in [6]-[8] discuss the Cloud and Fog computing protection from a data security perspective.

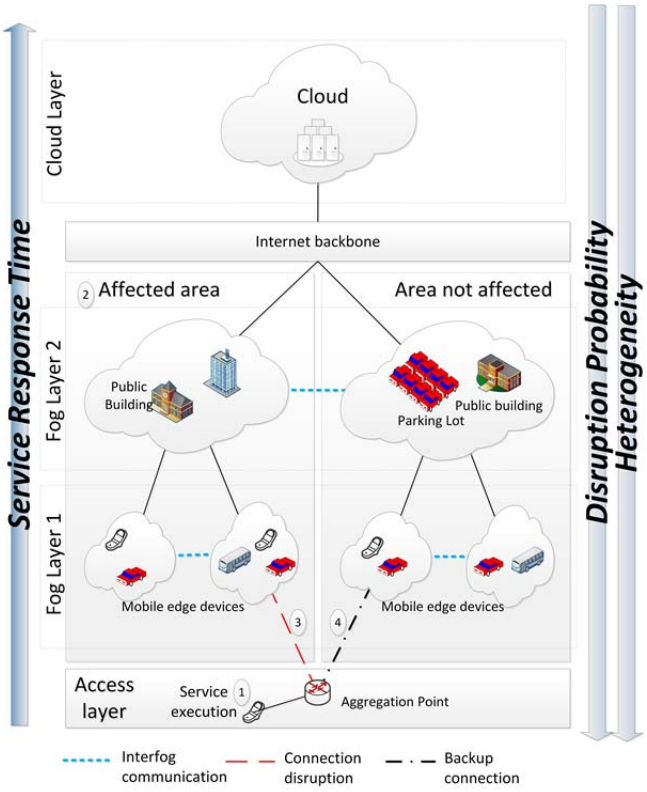


Fig. 1. Fog-to-cloud (F2C) topology.

These studies put major focus on the integrity and protection of the IoT data processed by the computing commodities.

The work in [9] presents a strategy for failure recovery in Mobile Edge Computing (MEC). The proposed strategy offloads workload to neighbor MEC upon fail occurrence or current MEC overload. However, no protection scheme is proposed to guarantee the availability of resources on neighbor nodes. Authors in [10] discuss characteristics of distinct network resilience strategies including protection, where backup resources are reserved in advance, and restoration, where resources are allocated after failure occurrence. Although the fact that protection schemes yield lower delay recovery, restoration schemes are often preferred due the efficiency on resources allocation. However, we consider that protection strategies must be assessed in F2C systems, whose applications often require low delay recovery.

The work presented in [5] is the first one to assess protection strategies in F2C computing systems. In that work, proactive and reactive strategies are confronted in order to illustrate pros and cons of each one. However, protection resources are always reserved in the same F2C layer of the resources actually allocated for service execution. Albeit benefits may be perceived in the proposed approach, it is worth assessing inter-layer protection strategies.

III. Modeling the PSA Problem

In this section, we introduce a topology model for the F2C architecture, the set of scenarios to be protected and, finally, we provide the analytical model for the PSA problem.

A. Topology Model

As shown in Fig. 1, the evaluated F2C topology consists of both Fog and Cloud commodities, all hierarchically distributed into three distinct vertical layers. The layer distribution is determined by the capacity, vicinity, and reachability to end-users consuming the IoT service. We assume the following topological assessments:

- End-users to be connected through two distinct Internet Service Providers (ISPs), one serving as the primary access and the other one as backup.
- The network infrastructure is owned by ISPs, but Cloud and Fog commodities do not.
- Cloud premises can be reached through any of the ISPs. However, certain Fog resources can only be reached through a certain ISP. Recall that not all nodes of an access layer belong to the same ISP. Indeed, we consider that the envisioned F2C layer distribution fits both commercial and networking availability requirements of end-users and ISPs.
- Each fog layer consists of several fog domains, hereinafter referred to as fog nodes. It is worth mentioning that the fog node concept is not yet clearly defined in the literature, and we are not going deep into this discussion. Rather, in this paper, we use the name fog node as a fog domain.
- In this paper, we deal with fog node failures (low granularity) as we assume that failures of individual edge-devices (high granularity) demands distinct protection strategies.
- Cloud providers implement their own resilience schemes so that the strategies employed in this work consider only failures in Fog nodes.

From a top-down perspective, the first layer is the Cloud layer. The Cloud layer provides near unlimited computing and storage resources, but with a significant cost in service transmission time. The next layer is the so-called Fog layer 2, or simply Fog-2. This layer embraces mobile as well as static edge-devices, providing resource aggregation in a neighborhood wide area, such as vehicles in a parking lot, or a set of buildings. Fog-2 nodes offer medium capacity and lower service transmission time in comparison with Cloud. The last layer is the so-called Fog layer 1, or simply Fog-1, which solely embraces mobile edge-devices, geographically close to end-users, which results in low allocation time. However, Fog-1 nodes show limited capacity and computing resources in comparison with upper layers in the hierarchy. As it can be seen in Fig. 1, 2 and 4 fog nodes are included in Fog-2 and Fog-1, respectively.

It is intuitive the clear interest in shifting services execution from the Cloud to the Fog layers in order to lower their transmission time. In addition, there are some concerns among ISPs related to the increase on bandwidth motivated by executing services at cloud. On the other hand, however, the execution of services solely at the fog layer is undoubtedly affected by the volatility inherent to devices at the edge—

motivated for example by energy saving policies or the intrinsic mobility of the edge devices.

It is worth highlighting the wide set of potential use-cases the F2C architecture may contribute to. Consider for example a medical assistance service in a smart city, where fog devices can change traffic lights color to accommodate the emergency squad to reach quickly the occurrence spot, whereas in a parallel manner the Cloud can look for emergency medical services discovery and location. A different example deserving specific attention focuses on ISP traffic offloading. Indeed, an ISP provider may take real-time decisions matching real-time resources availability in the F2C architecture to offload traffic among the different layers, thus enabling traffic load balance while minimizing the traffic to be handled by the network core.

In this paper, we consider two failure scenarios: 1) failure of one single fog node, where any fog node may become inaccessible in Fog layer 1 or Fog layer 2; and 2) general failure in one area (see Fig. 1), affecting one fog layer 2 and all fogs layer 1 directly connected to the compromised fog in layer 2 (hierarchical failure). For instance, as shown in Fig. 1, during the service execution (1), a hierarchical failure (2) may cause disruption on the service execution (3), however, the employment of protection resources (4) may guarantee its accomplishment. On the other hand, for the sake of simplicity, we assume cloud service providers to handle failures through the proper protection mechanisms. Therefore, in this paper we focus on the resilience of fog nodes.

B. Protection Strategies for F2C scenarios

In this subsection, we present the different protection strategies evaluated in this paper.

1) Horizontal Protection: In this strategy, the resources devoted for protection are allocated at the same F2C layer running the service. The deployment of a horizontal protection eases the recovery of services execution in resources with similar computing capabilities. In this way, service-level agreements are more highly to be respected in case of failures events.

2) Vertical Protection: This protection strategy allocates computing resources for protection at the immediately higher F2C layer. This strategy focus on diminishing the average delay for service allocation by employing the lowest fog layer exclusively for primary allocation.

3) Hybrid Protection: In the hybrid strategy, the selection of computing resources for protection is not dependent of the layer organization. The rationale behind this approach is to reduce the resource underutilization perceived in vertical protection strategy. For instance, in the topology shown in Fig. 1, idle fog resources in layer 1 could be used for protection in low load scenarios, whilst, in high load scenarios, protection could be allocated in higher layer resources, releasing lower latency resources for primary allocation.

4) Hierarchical Horizontal Protection: This strategy is designed to withstand failures affecting a local area, where neighbors Fog-1 and Fog-2 nodes are unavailable for service execution. Similar to horizontal protection, the computing resources selected for protection are positioned in the same layer of the ones selected for service execution, but, distinct from that strategy, hierarchical protection is placed exclusively on fog nodes positioned in a distinct area (area not affected, in Fig. 1).

5) Hierarchical Vertical Protection: This strategy is also suitable for dealing with local area failures. Similar to vertical protection, computed resources devoted for protection are located at higher layers. However, in this strategy, the amount of protection slots reserved in the cloud must be enough for the allocation of all primary resources used in one area.

We must highlight that the protection strategies so-called horizontal, vertical and hybrid protection, assume the failure of one computing node not to affect other nodes connectivity (these strategies are designed to deal with single-failure events affecting solely one computing node, hence not handling networking infrastructure or several computing nodes failures). However, hierarchical protection strategies are designed for topology configurations where the failure of a critical networking node, such as a gateway router or the multiple-failure of computing nodes, disrupts the execution of services on a particular area embracing both Fog-1 and Fog-2 nodes.

C. PSA Problem Mathematical Model

In this section, we model the PSA problem as a Multidimensional Knapsack Problem (MKP), presenting model details in a comprehensive manner. The model objective is two-fold: 1) reducing the latency for service allocation; and 2) reducing the P_{cost}. Therefore, the objective function (1) minimizes the sum of the allocation delay for each service in the set S as well as the sum of the recovery delay offered by each F2C resources in the set R according to the amount of resources employed for protection purposes. All symbols employed in the model are described in Table 1.

TABLE I. MODEL SYMBOLS DEFINITION

Symbol	Definition
D_i	Transmission delay related to service *i*, considering just the primary slots
P_r	Transmission delay related to service recovery using protection slots of resource *r*
S	Set of services to be executed
U_i	Total number of slots required to run service *i*
R	Set of F2C resources, i.e., set of cloud and distinct fog nodes
K_r	Set of slots provided by F2C resource *r* for both execution and protection
L_n	Set of fog nodes available on Fog layer *n*
H_r	Set of fog nodes positioned immediately under F2C resource *r* considering F2C hierarchy
F_r	Set of fog nodes in the same fog layer of fog *r*
G_r	Set of all F2C resources in the layer immediately above fog *r*
T_r	Allocation delay of a slot in F2C resource *r*, according to its F2C layer

Min: $$\sum_{i \in S} D_i + \sum_{r \in R} P_r \quad (1)$$

It must be remarked that the total computing capacity of both cloud and fog nodes are measured in terms of slot units. The so-called primary slots are the slots allocated for service execution, whereas the secondary slots are the slots selected for protection purposes. We consider that the minimum amount of slots required by a service is 1 slot unit. However, a service requiring more than 1 slot unit may be allocated in distinct F2C nodes and even in distinct F2C layers, as long as a F2C node has enough capacity, i.e., a service can be allocated regardless the service type.

It is worth noticing that services are executed in parallel. Therefore, in this scenario, a service execution delay is equal to the delay of the service slot allocated in the higher F2C layer, i.e., the one with the highest transmission delay.

The allocation of primary and secondary slots is respectively modeled as:

$$Y_{i,r,k} = \begin{cases} 1, & \text{if service } i \text{ is allocated in resource } r \\ & \quad \text{consuming primary slot } k \\ 0, & \text{otherwise} \end{cases}$$

$$X_{r,k} = \begin{cases} 1, & \text{if resource } r \text{ has its slot } k \text{ reserved as} \\ & \quad \text{a protection slot} \\ 0, & \text{otherwise} \end{cases}$$

Moreover, the following constraints are assumed. The total transmission delay related to primary slots is defined by (2). The total transmission delay related to protection slots (recovery delay) is defined by (3), where the overall protection delay of each fog-1 or fog-2 node equals the number of slots consumed for protection multiplied by the delay of a single transmission to this node.

$$\sum_{r \in R} \sum_{k \in K_r} Y_{i,r,k} * T_r = D_i \,, \forall i \in S \quad (2)$$

$$\sum_{k \in K_r} X_{r,k} * T_r = P_r \,, \ \forall r \in R \quad (3)$$

Constraint (4) is responsible for assuring the complete allocation of each service in the set S. It must be noticed that this constraint takes into consideration only the complete allocation of primary resources. Therefore, the total amount of primary slots allocated in both cloud and fog resources for each service must be equal to the number of slots required by the respective service.

Constraints (5) and (6) define the exclusivity and capacity constraints. The former ensures that each consumed slot is allocated as either one primary slot or one secondary slot, whilst the latter ensures that a computing (resource) node can be allocated for service execution or protection whenever it has at least one available slot.

$$\sum_{r \in R} \sum_{k \in K_r} Y_{i,r,k} = U_i \,, \ \forall i \in S \quad (4)$$

$$\sum_{i \in S} Y_{i,r,\mathrm{k}} + X_{r,\mathrm{k}} \leq 1 \,, \ \forall r \in R \ \wedge \ \forall k \in K_r \quad (5)$$

$$\sum_{i \in S} \sum_{k \in K_r} Y_{i,r,k} + \sum_{k \in K_r} X_{r,k} \leq |K_r| \,, \ \forall r \in R \quad (6)$$

Moreover, some specific constraints are defined to model properly the distinct protection strategies described in Section III-B. For horizontal protection, (7) is considered. It ensures the availability of protection slots in the same layer for any protected slot in Fog-1 or Fog-2, respectively represented by L_1 and L_2. In addition to this equation, the hierarchical horizontal protection strategy is modeled by (8). This equation ensures the availability of the protection slots for recovery in case of a failure affecting primary slots allocated in both Fog-2 and Fog-1, i.e., a failure affecting the network connectivity to both Fog-1 and Fog-2, or a multiple-failure scenario affecting both Fog-1 and Fog-2 nodes.

$$\sum_{i \in S} \sum_{k \in K_r} Y_{i,r,k} \leq \sum_{q \in F_r - \{r\}} \sum_{k \in K_q} X_{q,k} \,, \quad \forall r \in L_1 + L_2 \quad (7)$$

$$\sum_{i \in S} \sum_{q \in H_r} \sum_{k \in K_q} Y_{i,q,k} \leq \sum_{q \in L_1 - H_r} \sum_{k \in K_q} X_{q,k} \,, \quad \forall r \in L_2 \quad (8)$$

Focusing on the vertical protection strategies, (9) ensures that the amount of protection slots in a layer is enough for the recovery of any F2C node failure in the layer below. Moreover, (10) ensures the availability of protection slots for hierarchical vertical recovery of any Fog layer 2 node failure.

$$\sum_{i \in S} \sum_{k \in K_r} Y_{i,r,k} \leq \sum_{q \in G_r} \sum_{k \in K_q} X_{q,k} \,, \quad \begin{matrix} \forall r \in L_1 + L_2 \,, & \text{if vertical strategy} \\ \forall r \in L_1 \,, & \text{if hierarchical vertical strategy} \end{matrix} \quad (9)$$

$$\sum_{i \in S} \sum_{q \in H_r + \{r\}} \sum_{k \in K_q} Y_{i,q,k} \leq \sum_{q \in G_r} \sum_{k \in K_q} X_{q,k} \,, \quad \forall r \in L_2 \quad (10)$$

Another protection strategy implemented in this work consists in a hybrid of horizontal and vertical protection strategies. This approach is modeled by (11), whose priority is the allocation of primary slots in resources located in lower layers (for reducing the transmission delay) but also allowing the use of spare resources in the same layer for protection slots.

$$\sum_{i \in S} \sum_{k \in K_r} Y_{i,r,k} \leq \sum_{q \in R} \sum_{k \in K_q} X_{q,k} \,, \quad \forall r \in L_1 + L_2 \,|\, q \neq r \quad (11)$$

IV. Numerical Results

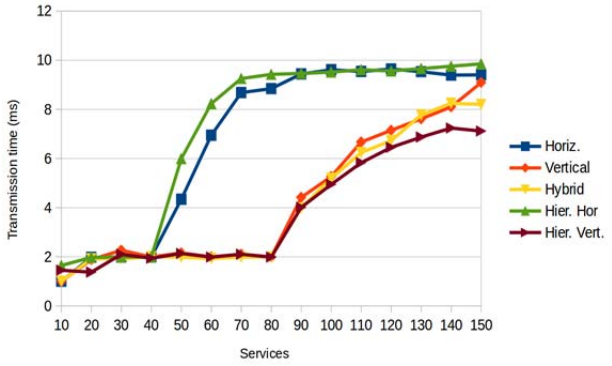
In this section, we present numerical results related to the evaluation of both transmission delay and P_{cost}. All plotted values have a 95% confidence interval. The presented numerical results were obtained by means of the well-known optimization tools PuLP [11] and Gurobi Optimizer [12]. In addition, Tables 2 and 3 summarize the set of simulation parameters used in the trials.

TABLE II. Simulation Parameters: Services

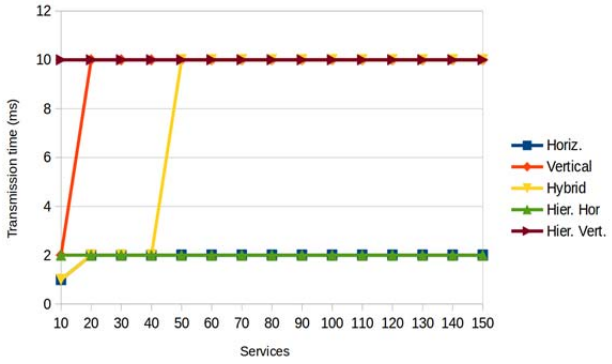
Parameter	Value
Number of requested services	From 10 to 150
Ratio between amount of services: low consuming (mice) / high consuming (elephants)	90 (low) / 10 (high)
Slots demanded by mice	3
Slots demanded by elephants	30

TABLE III. SIMULATION PARAMETERS: F2C RESOURCES

Parameter	Fog-1	Fog-2	Cloud
Number of F2C nodes per layer	4 fogs	2 fogs	1 cloud
Available resources per F2C node	20	200	Nearly unlimited
Transmission delay per F2C layer	1 ms	2 ms	10 ms



(a)



(b)

Fig. 2. Average service transmission delay: (a) primary slots; (b) protection slots.

Fig. 2 illustrates the impact on the average service transmission delay for each PSA strategy. As it may be seen in Fig. 2(a), the deployment of horizontal protection strategies leads to a substantial increase on the average service transmission delay in contrast to hybrid and both vertical protection strategies even for a relatively low number of services. On the other hand, the delay related to protection slots (recovery delay), shown by Fig. 2(b), uncovers the tradeoff perceived on protection strategies. Albeit horizontal protection strategies cannot offer low impact on resource allocation delay for a medium to high number of services, the recovery delay is considerably lower than other strategies, even with a high amount of services. In fact, the increasing number of services does not affect the protection latency in horizontal strategies since the amount of protection slots in lower fog layers will be enough for the recovery of any fog node failure, whereas the premature extinguishment of free primary resources in lower layers enforces the allocation of new services in the cloud, increasing the primary allocation delay, as seen in Fig. 2(a). However, vertical protection strategies offer low impact on the primary delay at the cost of a high latency recovery, i.e., the delay related to the protection slots.

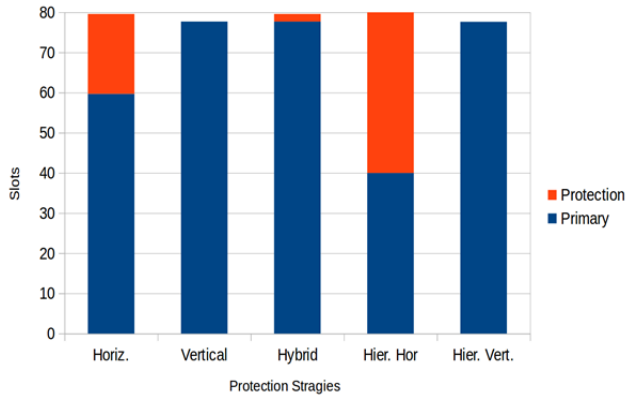


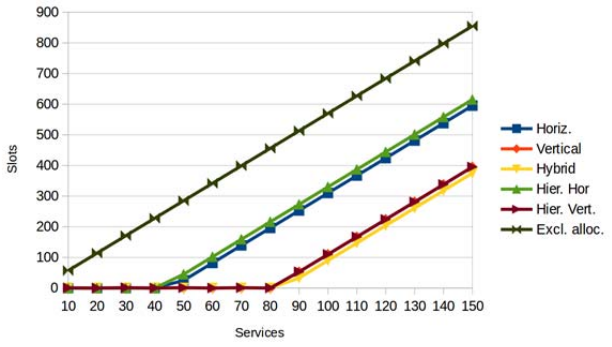
Fig. 3. Average resource allocation distribution in Fog-1.

It is worth noticing that the primary delay observed in the hybrid strategy is comparable to the one offered by vertical protection strategies. However, the service recovery delay is significantly affected by the amount of service requests.

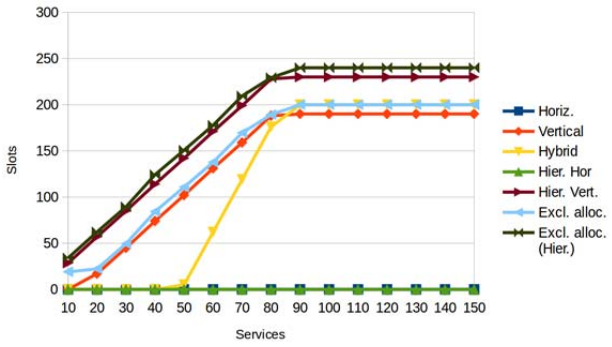
The high recovery latency observed in vertical strategies is explained through the analysis of resource allocation in the lowest fog layer (the one more vulnerable to congestion and lack of computing resources), depicted by Fig. 3.

One can notice that all resources are allocated as primary slots in vertical strategies even when there are free resources available in this layer, inhibiting the low delay recovery, whereas the hybrid strategy makes use of spare resources for protection purposes. Moreover, this figure shows that the resource utilization in Fog-1 can be as low as 50% for hierarchical horizontal strategies, whereas a horizontal strategy achieves a utilization ratio of 75%, also giving ground for the previously presented analysis regarding the low recovery delay in horizontal strategies enabled by high availability of protection resources in lower layers.

On the other hand, Fig. 4 shows cloud resources allocation under distinct protection strategies. Fig. 4(a) is devoted to illustrate the primary resource allocation. For sake of comparison, we include the amount of allocated slots for an exclusive cloud allocation strategy, which consists in the employment of cloud as the unique available resource for service execution—traditional cloud scenarios. As may be noticed by the reader, as the employment of horizontal strategies increases the employment of fog resources for protection purposes, more cloud resources must be used for primary slots allocation in comparison to vertical and hybrid strategies. In contrast, the analysis of Fig. 4(b), where the amount of resources employed in cloud resources for protection purposes is represented, shows that horizontal strategies have minimum load in cloud resources. In fact, this is explained by the assumption that cloud providers implement their own resilience strategies. Once more, we added exclusive cloud allocation results in this figure, where the first—Excl. alloc.—is the allocation of secondary slots demanded by the recovery of any single fog failure, while the second—Excl. alloc. (Hier.)—considers the allocation of secondary resources for one area failure recovery, as earlier discussed in this paper.



(a)



(b)

Fig. 4. Resource allocation distribution in the Cloud: (a) primary slots; (b) protection slots.

Based on the obtained results, we can see that albeit F2C can significantly reduce the amount of resources consumed in higher layers and the service transmission time, distinct protection strategies must be considered for distinct scenarios aiming at the employment of the most suitable one, regarding to specific demands of each scenario. For instance, fog nodes providing sensitive services, such as healthcare and Intelligent Transportation Systems (ITS), might leverage horizontal strategies to ensure low delay recovery, whereas fog nodes supporting no sensitive applications with high resource consuming, such as video streaming, may make use of vertical or hybrid strategies. In both cases, services can be executed in a reliable way, not overusing cloud resources.

V. CONCLUSION

Fog-to-Cloud architectures (F2C) have emerged as an innovative technology to fulfill the ever increasing requirements of IoT scenarios. In this work, we put the focus on studying distinct protection techniques aimed at increasing the resilience level of F2C architectures. To this end, we present an extensive evaluation related to the minimization of both service transmission delay and protection cost in protection scenarios, and important factors for obtaining low latency in reliable IoT service execution. The presented results showed that the selection of a protection strategy has a substantial impact on the service execution and resilience performance.

As a future line of work, we plan to study how the service discovery process is affected when F2C resilience is considered as well as to evaluate the impact of power consumption on failure scenarios.

ACKNOWLEDGMENTS

This work was supported by the Spanish Ministry of Economy under contract TEC2015-66220-R, by the Catalan Government under contract 2014SGR371 and by the H2020 EU mF2C Project ref.730929. Vitor Barbosa Souza is supported by CAPES Foundation, Ministry of Education of Brazil, Proc. No. 11888/13-0.

REFERENCES

[1] L. Heilig and S. Voss, “A Scientometric Analysis of Cloud Computing Literature,” In: IEEE Transactions on Cloud Computing, vol. 2, no. 3, pp. 266–278,July 2014.

[2] F. Bonomi, R. Milito, J. Zhu and S. Addepalli, “Fog Computing and Its Role in the Internet of Things,” in Proceedings of 1st MCC Workshop on Mobile Cloud Computing, ser. MCC ’12. New York, NY, USA: ACM, 2012, pp. 13–16. doi: 10.1145/2342509.2342513.

[3] X. Masip-Bruin, E. Marín-Tordera, G. Tashakor, A. Jukan and G. J. Ren, “Foggy clouds and cloudy fogs: a real need for coordinated management of fog-to-cloud computing systems,” in IEEE Wireless Communications, vol. 23, no. 5, pp. 120-128, October 2016. doi: 10.1109/MWC.2016.7721750.

[4] C. Develder, J. Buysse, B. Dhoedt, and B. Jaumard, “Joint dimensioning of server and network infrastructure for resilient optical grids/clouds,”. Iin: IEEE/ACM Transactions on Networking (TON), vol.22, no. 5, pp. 1591-1606, October 2014. doi: 10.1109/TNET.2013.2283924.

[5] V. B. Souza, X. Masip-Bruin, E. Marín-Tordera, W. Ramírez and S. Sánchez-López, “Proactive vs Reactive Failure Recovery Assessment in Combined Fog-to-Cloud (F2C) Systems,” in 2017 IEEE 22th International Workshop on Computer Aided Modelling and Design of Communication Links and Networks (CAMAD), Lund, 2017.

[6] J. K. Liu, K. Liang, W. Susilo, J. Liu and Y. Xiang, “Two-Factor Data Security Protection Mechanism for Cloud Storage System,” in IEEE Transactions on Computers, vol. 65, no. 6, pp. 1992-2004, June 1 2016. doi: 10.1109/TC.2015.2462840.

[7] Y. Li, W. Dai, Z. Ming and M. Qiu, “Privacy Protection for Preventing Data Over-Collection in Smart City,” in IEEE Transactions on Computers, vol. 65, no. 5, pp. 1339-1350, May 1 2016. doi: 10.1109/TC.2015.2470247.

[8] S. J. Stolfo, M. B. Salem and A. D. Keromytis, “Fog Computing: Mitigating Insider Data Theft Attacks in the Cloud,” 2012 IEEE Symposium on Security and Privacy Workshops, San Francisco, CA, 2012, pp. 125-128. doi: 10.1109/SPW.2012.19.

[9] D. Satria, D. Park and M. Jo, “Recovery for Overloaded Mobile Edge Computing,” in Future Generation Computer Systems, vol. 70, pp. 138–147, May 2017, http://dx.doi.org/10.1016/j.future.2016.06.024.

[10] B. Mukherjee, M. F. Habib and F. Dikbiyik, “Network adaptability from disaster disruptions and cascading failures,” in IEEE Communications Magazine, vol. 52, no. 5, pp. 230-238, May 2014. doi: 10.1109/MCOM.2014.6815917.

[11] Optimization With PuLP. Available: http://www.coin-or.org/PuLP/.

[12] Gurobi Optimization. [Online]. Avaliable: http://www.gurobi.com/.