DiVA
http://www.diva-portal.org

Preprint

This is the submitted version of a paper presented at *SEAA 2019 : 45th EUROMICRO SEAA Conference, Aug 28, 2019 - Aug 30, 2019, Thessaloniki / Chalkidiki, Greece.*

Citation for the original published paper:

Bakhshi Valojerdi, Z., Rodriguez-Navas, G., Hansson, H. (2019)
Dependable Fog Computing: A Systematic Literature Review
In:

N.B. When citing this work, cite the original published paper.

Permanent link to this version:
http://urn.kb.se/resolve?urn=urn:nbn:se:mdh:diva-45152

Dependable Fog Computing: A Systematic Literature Review

Zeinab Bakhshi[1], Guillermo Rodriguez-Navas[2], Hans Hansson[1]
[1] Mälardalen University, Sweden, {zeinab.bakhshi, hans.hansson}@mdh.se
[2] Nokia Bell Labs, Israel, guillermo.rodriguez-navas@nokia-bell-labs.com

***Abstract*— Fog computing has been recently introduced to bridge the gap between cloud resources and the network edge. Fog enables low latency and location awareness, which is considered instrumental for the realization of IoT, but also faces reliability and dependability issues due to node mobility and resource constraints. This paper focuses on the latter, and surveys the state of the art concerning dependability and fog computing, by means of a systematic literature review. Our findings show the growing interest in the topic but the relative immaturity of the technology, without any leading research group. Two problems have attracted special interest: guaranteeing reliable data storage/collection in systems with unreliable and untrusted nodes, and guaranteeing efficient task allocation in the presence of varying computing load. Redundancy-based techniques, both static and dynamic, dominate the architectures of such systems. Reliability, availability and QoS are the most important dependability requirements for fog, whereas aspects such as safety and security, and their important interplay, have not been investigated in depth.**

I. INTRODUCTION

Fog computing is a recently introduced computational paradigm that extends cloud resources closer to the edge of the network [1][2]. The so-called fog lies between the cloud and edge and aims at providing efficient data processing, effective analysis and storage capacity. In addition, it reduces the amount of data transmitted to the cloud [3]. Most authors agree on a 3-layer architecture for fog computing, as depicted in Figure 1, which encompasses cloud resources on the top (Cloud layer), fog nodes in the middle (Fog layer) and edge devices at the bottom, in the Sensor layer. More fine-grained architectures have been proposed, such as the Openfog Reference Architecture published by the Openfog Consortium [4], but there is still no general consensus about which elements should be included in each layer.

The Fog layer brings notable improvements in terms of reduced latency, increased performance, scalability and adaptability. However, it also faces a number of challenges, due to its distributed and open nature, which are not present in the cloud paradigm, such as the presence of unreliable wireless links, energy-constrained edge nodes, congestion and scalability issues associated to adaptive services like application placement, task allocation and network configuration [1]. In addition to potential security threats related to remote untrusted nodes and wireless communication.

Despite these difficulties, dependability has not played a central role in the design of fog computing solutions, which might prevent the future adoption of this technology for critical applications. In order to understand the current situation, future trends and open gaps, in this paper we perform a systematic literature review (SLR) about dependability and fog computing. SLR is a procedure that involves identification and analysis of both quantitative and qualitative evidence in response to a given research question, such that an empirical yet rigorous and transparent answer can be provided [5].

Fig. 1. Fog computing scheme

The remainder of this paper is structured as follows: We describe classical dependability notions, in Section II. We continue with a description of the applied research methodology in Section III. In Section IV, we analyze the extracted data, according to our technical classifications, publication and contribution information. Limitations in this study and threats to validity are described in Section V. Research gaps and future research directions are discussed in Section VI. Finally, we conclude the study in Section VII.

II. CLASSICAL DEPENDABILITY NOTIONS

Dependability is the ability of a system to provide its desired service continuously over a defined period of time; a dependable system is a system that is trusted and available. The works of Avizienis and Laprie remain as the cornerstones of this research area; see for instance [6] and the references within. In this work, we will adhere as much as possible to the classical definitions of dependability.

As Figure 2 shows, dependability can be divided into three classes of notion (extracted from [7], Chapter 3). The first class of concept for dependability is the dependability *attributes*. Attributes are the main requirements of the system

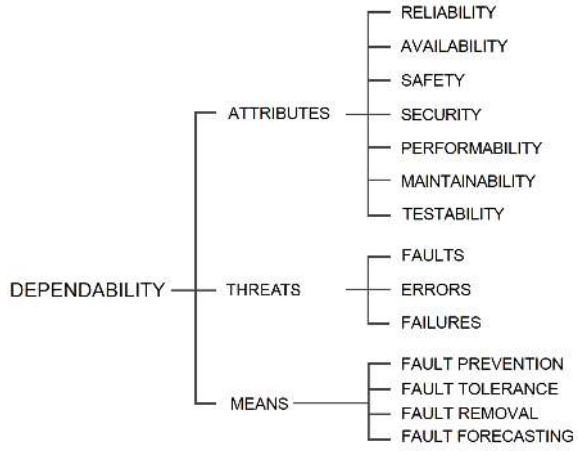

Fig. 2. Classes of notion for dependability [7]

to be dependable, and they can be measured using quantitative or qualitative metrics. The meaning of the attributes are as follows (we refer the interested reader to [6] for a more detailed description):

- Reliability: continuity of correct service.
- Availability: readiness for correct service.
- Serviceability: the ability to provide required service.
- Security: the ability to keep services running over malicious faults.
- Safety: the state of being safe from harm or other non-desirable outcome.
- Performability: a measurement of how a system performs over time.
- Maintainability: ability to undergo modifications and repairs.
- Testability: a degree in which a service supports testing.

The second concept is *threats* (or impairments) to dependability, which are the undesired events that cause the system to fail to perform its service. Fault, error and failure form a chain connected together. Fault is a defect of one of the system's components and may cause errors in the system; an error is an incorrect value in the system state, which may result in a deviation of the desired service - a failure.

The third class of notion is the *means* of dependability. Means are the methods and techniques that can be used to ensure a system is dependable and can deliver the defined service. To have a dependable system a combination of these means should be considered from the design stage of the system.

- Fault Prevention: means to prevent the occurrence or introduction of faults;
- Fault Tolerance: means to avoid service failures in the presence of faults;
- Fault Removal: means to reduce the number and severity of faults;
- Fault Forecasting: means to estimate the present number, the future incidence, and the likely consequences of faults.

III. Research Method

To conduct our systematic literature review we applied the method proposed in [5], [8] and [9]. The process diagram of this research method is outlined in Figure 3 and we will describe each step in the following subsections.

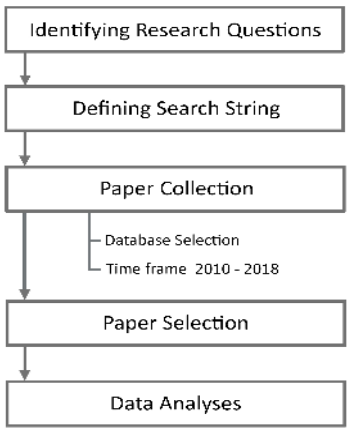

Fig. 3. Research Method

A. Identifying the Research Questions

To make the goals of our study explicit, we define the following research questions (RQs).

- RQ1 Which classical dependability attributes do authors consider as relevant for fog computing?
- RQ2 Which are the considered sources of failure in fog computing?
- RQ3 What dependability means are applied to ensure dependability in fog computing?
  - RQ3-1 What specific methods/techniques are adopted in order to implement these dependability means?
- RQ4 What is the relation between dependability and security in the solutions proposed for fog computing?

The result of this study are the answers to the above questions based on analysis of each paper identified in our selection and classification.

B. Defining the Search String

The goal of the search is to identify all papers that address fog computing and dependability. To that aim, we define a search string based on keywords that are related either to fog computing or to the dependability attributes. To be inclusive, the search string should also include terms that are synonyms, or fairly similar, to fog computing. For instance, we realized that the terms fog computing and edge computing are often used in ambiguous ways, almost interchangeably. We also discovered that some authors prefer the term cloudlet. Adding all these terms in our search string resulted in a high number of papers retrieved in the first phase.

For dependability attributes, we decided to use Avizienis' reference dependability tree [6], explained in Section II.

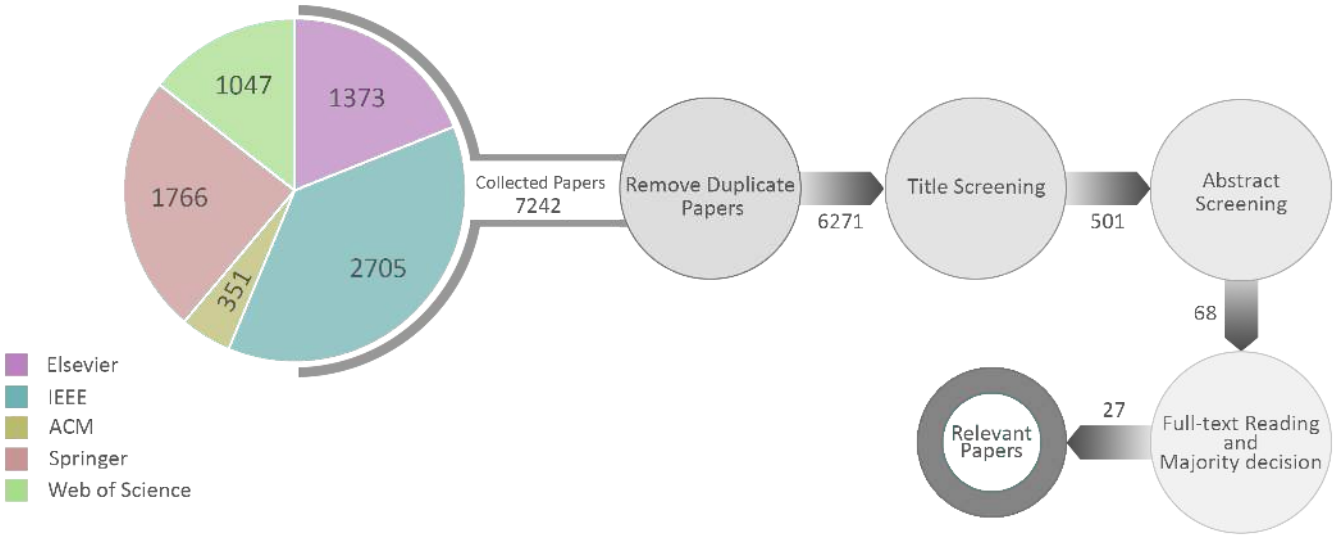

Fig. 4. Paper Selection Process

We used the Boolean operator OR to join synonyms and keywords and the AND operator for our two main terms, fog computing and dependability attributes. Therefore, the complete search string includes two parts and is formulated as:

(Fog OR fog computing OR edge computing OR cloudlet) AND (Reliability OR dependability OR availability OR serviceability OR security OR safety OR performability OR maintainability OR testability)

We used this search string to collect papers in our selection of databases, which is described in the next subsection.

C. Collecting Papers

To collect papers for this study, first we identified the five most popular scientific online digital libraries: 1) Web of Science; 2) Science Direct; 3) IEEE Xplore; 4) Springer and 5) ACM.

We searched our search string in each database of aforementioned digital libraries. We had to identify the correct time frame for the search. Our approach was to take the original publication of Cisco in 2012 [1] (in which the term fog computing was coined) as the starting point, but we extended the search to two years earlier just to make sure that we were not missing a previous publication including terms similar to fog computing.

After determining databases for references and defining the date range, we performed a search for available papers matching our criteria. In our initial search we collected 7242 papers in total from all mentioned databases.

D. Selecting papers

In this step we filtered out irrelevant papers. Different levels of filtering were applied sequentially to the collection of papers. This phase is one of the most time-consuming and difficult parts in a systematic literature review [10]. Figure 4 shows the different stages of this phase. We used Mendeley and Endnote to manage papers' records and removing duplicates respectively. Among 7242 papers found in the databases, we filtered 6271 papers after removing 971 duplicate papers. For the remaining 6271 papers, we filtered relevant papers in three steps, first, title screening; second, abstract screening and, finally, the third step was full-text reading and majority decision making.

In the title screening phase we made two categories: Irrelevant and Relevant. After this step, more than 80% of papers were filtered out and placed into the Irrelevant category. As a result, we had 501 Relevant papers selected for abstract screening. The result of the abstract screening, which included the same categories, was that 68 papers were marked as Relevant for the next step (full-text reading). The criteria applied for the selection is that a paper was marked as a Relevant study if the Abstract contained both some terms relevant to fog and some terms related to dependability, such as attributes, threats or means.

The abstract screening was slightly optimistic, in the sense that we allowed papers containing some ambiguities to pass to the next phase. This is justified by our interest in identifying all relevant papers, but it forced us to introduce an additional review effort, namely expert reading, which turned out to be a major decision in terms of efforts required. In particular, we noted that the terms reliability and safety are used in a broad sense, often ambiguous, to refer to aspects such as quality, performance or accuracy, which are not specific to dependability. These papers were filtered out during the full text reading. It was also found that some papers referred to the dependability attributes correctly and emphasized their importance for fog, but the full text reading did not provide any evidence that these attributes were actually addressed at all. Such papers were also eliminated.

As a consequence, in the final step after full-text reading and majority decision only 27 papers remained as Relevant papers for this study. Upon detailed reading, we found that

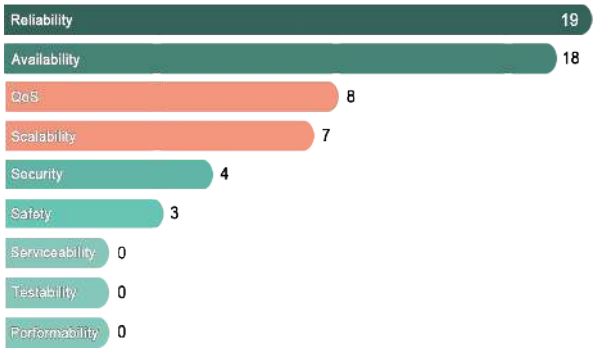

Fig. 5. Frequency of the Observed Dependability Attributes

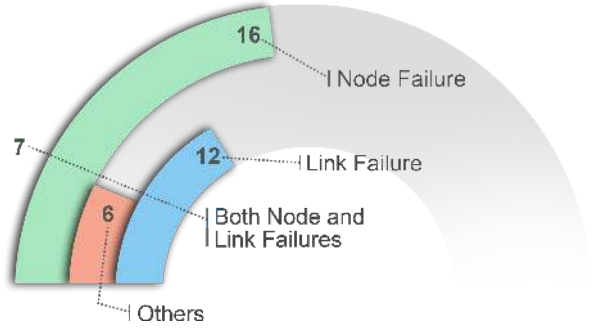

Fig. 6. Source of Threats

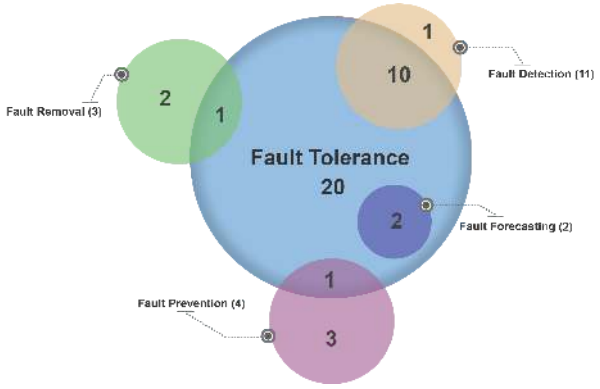

Fig. 7. Observed Dependability Means

papers [11] and [12] are similar to a large extent considering the underlying theme, model and approach, which can have a slight bias on the study results. However, since [12] looks like a journal version of [11] with improvements, we have decided to keep both of them, to be consistent with our predefined selection rules

IV. Technical Data Analysis

Data extracted from relevant papers are analyzed based on four main categories: 1) Dependability Attributes; 2) Source of Threats 3) Dependability Means (Fault, Failure and Errors) and 4) Threat Detection and Response Methods. According to these categories, Table I shows a summary of different dependability concepts considered in each paper.

A. Dependability Attributes

Figure 5 illustrates the number of papers focusing on each dependability attribute. In addition, it is observed that in almost 30% of papers, Quality of Service (QoS) is the main focus. Scalability, which in principle is not part of the dependability taxonomy, has attracted authors' attention as an attribute to make fogs dependable, because of the increasing number of connected devices. We decided to include these two attributes in the study, in addition to the traditional dependability attributes. Although it seems that safety is an important subject for adapting fog computing in critical application areas, we noticed that only 11% of papers focused on safety aspects in fog computing. This might be explained by the little industrial adoption of fog computing, since safety typically requires proven technologies.

B. Source of Threats

As discussed in Section II, fault, error and failure form a connected chain during system development and execution. In this section, we group all of them as threats, and we focus on identifying the reported source of these threats. We found that authors mostly focused on three main types of failures: node, link and "other" types of failures, which includes application placement failure, specification failure, late performance and failures due to resource constraints. The results are depicted in Figure 6. Note that there are 7 papers focusing on both node and link failure.

Node Failure: Fog nodes are considered as either physical hardware devices or virtual machine (VM) nodes. Failure of a fog node is discussed in terms of storage, memory, computation failure and data crashes. *Link Failure*: Link or path failure is the failure of the virtual and/or physical links in fog infrastructures. Different types of link failures are investigated, in different directions: 1) Cloud-to-fog links and vice versa; 2) Fog-to-fog links, and 3) Fog-to-edge devices link and vice versa. However, not always the link failure direction is explicitly specified in the studies. *Other types* of failure: Fog computing resources are responsible for providing services and placing these services or applications throughout the network. Services are shared and placed in fog nodes, edge network devices and other components that may require to update these services dynamically. Limitations in resource allocation, application placement, task scheduling, etc. may result in failure for providing the expected services. Whether this failure is caused by an inaccurate estimation of the workload or by an inaccurate estimation of the resource capacity is not discussed in the considered papers.

C. Dependability Means

Figure 7 shows how many studies focused on each of the dependability means. It is very clear that fault tolerance is the preferred and most investigated method. There is one paper, Smara et al. [32], which proposed a solution in which the system stops working upon fault detection, and stays in a safe state, without resorting to fault tolerance.

TABLE I

SUMMARY OF REVIEWED STUDIES

	Observed Dependability Attributes						Source of Threat			Threat Management				
Reviewed Studies	Availability	Reliability	Security	Safety	QoS	Scalability	Node Failure	Link Failure	Others	Fault Tolerance	Fault Detection	Fault Prevention	Fault Removal	Fault Forecasting
Aral & Brandi. [13]	✓	✓			✓		✓			✓	✓			✓
Benson et al. [14]	✓					✓		✓		✓	✓			
Breivold et al. [15]	✓	✓	✓	✓	✓	✓								
Fitzek et al. [16]	✓						✓						✓	
Cau et al. [17]	✓					✓		✓		✓	✓			
Chen et al. [18]	✓	✓					✓	✓		✓				
Chervyakov et al. [19]	✓	✓	✓	✓		✓	✓			✓	✓			
Dasari et al. [20]	✓						✓					✓		
Elbamby et al. [11]		✓							✓	✓	✓			
Elbamby et al. [12]		✓							✓	✓	✓			
Huang & Xu. [21]		✓			✓				✓			✓		
Itani et al. [22]	✓	✓					✓			✓				
Jonathan et al. [23]	✓	✓							✓	✓	✓			
Kumar et al. [24]		✓					✓	✓					✓	
Liu & Zhang [25]		✓						✓	✓	✓		✓		
Mennes et al. [26]	✓						✓	✓		✓	✓			
Okafor et al. [27]	✓	✓	✓		✓	✓		✓						
Osanaiye et al. [28]	✓	✓	✓		✓	✓	✓			✓				✓
Popentiu et al. [29]	✓	✓					✓	✓		✓				
Rimal et al. [30]	✓	✓			✓		✓	✓		✓				
Saqib & Hamid. [31]		✓					✓			✓				
Smara et al. [32]				✓		✓	✓				✓			
Sood. [33]	✓				✓				✓			✓		
Spinnewyn et al. [34]	✓						✓			✓				
Wiss & Forsstrom . [35]		✓			✓			✓		✓	✓			
Xiao et al. [36]	✓	✓					✓	✓		✓				
Zhou et al. [37]	✓						✓	✓		✓	✓			

This paper is the only one in the Fault detection category. Fault forecasting is related to estimating the faults potentially leading to resource allocation failures because of incorrect workload (or capacity) estimation. Fault prevention concerns the selection of resources that guarantee lower occurrence of faults.

D. Threat Detection and Response Methods

Providing a dependability solution is tightly bounded to threat detection and how to set the strategies to respond to these threats. We investigated different threat detection and threat response methods:

- Threat Detection methods: are in general supported by tools. As described in Figure 7, there are 11 papers in our study which introduce tools and methods for threat detection. Some examples are: heart beat protocol [16], self checker [32], genetic algorithm [26] and machine learning [13].
- Threat Response methods: the most common approach for responding to threats is redundancy. We observed that 77% (21 out of 27) of the considered studies used redundancy techniques and replication of sources or components as a response method for dependability solutions. Many different kinds of replication strategies are reported: Natural redundancy, Temporal redundancy and Spatial redundancy, including Active/Active, Active/Passive (primary/backup). Dynamic and Static solutions are found alike, and they are applied for responding to link, node and other types of failure.

This indicates that the applied techniques in considered papers do not differ significantly from standard practices [6] in other domains.

E. Publication and Contribution Data Analyzes

The justification for using research type as a dimension for analyzing the studies is to aid us to understand the maturity and weight of the research performed and, in that perspective, to better understand the research approaches in general. Our analysis considers the following three dimensions: 1) the application domain of papers; 2) the addressed fog computing service; 3) scientific research groups and 4) research methods and tools.

1) Fog computing application domains. Among the 27 papers, 3 papers are in vehicular application, [21], [36] and [31], which has a use-case in connected cars. We found another work mainly focusing on energy internet [18] and the rest are proposing their solution generally for IoT and Industrial IoT, or fog computing in general.

2) Addressed fog computing service. In general, the considered papers are mainly targeting one of two fundamental issues, (1) how to collect data from unreliable and non-trusted mobile nodes and (2) task offloading and task allocation, for instance task allocation from cloud resources to fog nodes. The solutions proposed for (1) are mostly redundancy techniques or sharing data and information with

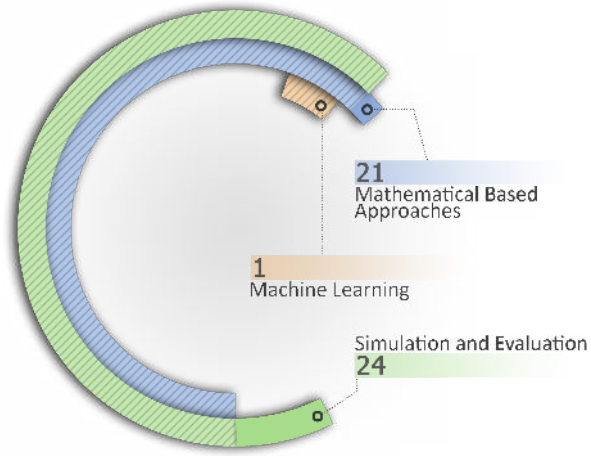

Fig. 8. Adopted methods and tools

neighbour nodes. For (2), authors proposed optimization, task monitoring and dynamic task allocation as possible solutions. In addition to these two main categories, there are a few papers with other focuses, which cannot be grouped under a common denominator.

3) Scientific research groups. The study of the authors' affiliations does not provide any significant insight, other than that the research is really spread worldwide and that no group has a predominant position. For the 27 papers we identified 50 different research groups distributed in 25 countries; with only two universities having two papers (Xiadian University in China and Universidad de Antioquia in Colombia).

4) Research Methods and Tools. More than 87% of the selected papers included some form of empirical verification or simulation. In addition to this, authors applied mathematical based methods for calculation or verification purposes in 77% of the considered papers. Only one study applied machine learning (ML) methods for fault detection and fault forecasting purposes. Figure 8 shows the number of studies using different methods for assessing the dependability solutions.

F. Discussion

The analysis presented in the previous subsections provide us with the evidence to answer the research questions defined in Section III.

Our answer to RQ1 is given in Figure 5. Reliability and availability are the dependability attributes that most authors are focusing on. In contrast, testability and performability are not considered at all in the considered literature. We observed that QoS and Scalability are new terms which attracted authors attention in ensuring dependability in fog computing. Security and Safety are discussed in 14% and 11% of papers, respectively. There is no single paper considering the interplay between safety and security.

The answer to RQ2 is based on Figure 6. Node failure and link or path failure are the main source of failures considered in the literature. We also observed that failures due to resource allocations and application placement are listed under a third category we call other types of failures.

RQ3 is answered by Subsection IV-D. We observed that redundancy techniques are the most common methods to increase dependability level in fog computing. The applied methods do not vary much with respect to other domains, although we noticed a lower presence of formal verification methods, probably caused by the limited interest in safety.

The answer to RQ4 is given by Figure 5. We found many papers related to security in fog computing but most of them are not among the papers that are relevant to our topic (dependability) and were filtered out in the abstract screening. Security in fog is usually discussed from the perspective of privacy and confidentially, with methods like encryption, identification and authentication. However, the existence of malicious faults and the design of fault-tolerant security solutions are not addressed in the considered papers. The only exceptions are the papers dealing with collecting data from untrusted mobile nodes.

We found that in the context of fog computing, it makes sense to extend the traditional dependability attributes with, QoS and scalability, which have received significant attention.

V. Limitations and Threats to Validity

There are several issues that need to be taken into account when conducting a systematic literature review. These issues can potentially limit the validity of obtained results. In the following we will list a number of limitations as well as threats to validity of this work.

A. Limitations

1) Large volume of data: . In this study, 7242 papers were collected initially and 6271 papers were selected for Title screening. This high number is caused by the confluence of several factors: the difficulty to discriminate (based on title) fog as a meteorological phenomenon from fog as a computing paradigm; the need to include publications referring to e.g. edge computing to make sure that other synonyms of fog were not missed.

2) Procedural limitations related to the databases: . Extracting search results from *IEEE Xplore* digital library: When the number of papers found with the search string exceeds 2000 papers, it is not possible to download all the citations from IEEE Xplore. So we divided our search from 2010 to 2015 and then from 2015 to 2018, and later on removed duplicates to have the results in two parts, each part consisting of less than 2000 papers. We contacted an IEEE search expert and reported this issue. Extracting search results from *Elsevier* digital library: Elsevier data base does not support more than 8 Boolean operators in a single search string. So we divided our search string into two strings to cover all papers within our complete search string.

B. Threats to validity of the Results

The research method in this work intends to capture all papers addressing dependability in fog computing, but it is

possible that a number of relevant papers are not detected because of the following reasons.

1) English-only publications: we only included papers that are written in English and there is a chance that we miss important papers written in other languages. This, we believe, is a limitation with most of the systematic literature reviews. 2) Digital copy of works: we decided to include studies that are available and published electronically. There is a chance that a relevant paper is not published online due to confidentiality or other reasons. Our systematic literature review does not extend to such scenarios. 3) No *snowballing*: in many Systematic literature reviews, there is a step in which the same process for paper selection can be executed for the references of selected papers, which is called snowballing. We did not go through this stage in this work. 4) Fog-related terms: our focus on fog-related terms could scope the search in a limiting way, i.e., there could very well be papers that deal with related issues, but that uses a different terminology, e.g. papers dealing with classical distributed system dependability.

Mistakes during the filtering of papers might have happened due to the following factors: 1) Poor abstract but rich content: there is a chance that we missed relevant papers which had a poorly written abstract yet were discussing issues important to our research questions. 2) Lack of adherence to the classical dependability taxonomy: the ambiguous use of some terms, most notably reliability and safety, made it difficult to identify papers that were actually addressing these aspects. To minimize the impact of this threat, we were optimistic in the selection of papers and introduced an additional review of the full text, performed independently by two experts.

We know of at least one paper that deals with fog computing and dependability [38], but which was not captured in our paper collection phase because it does not use any of the fog computing synonyms. Interestingly enough, the paper falls exactly within the categories detected in our study, since it addresses reliability in the allocation of tasks to nodes, and it uses fault prevention and static fault tolerance for the nodes, without considering link failures. This gives us increased assurance that even if a number of papers are missed, the results of the SLR are valid and can be generalized.

VI. Research Gaps

The current study provided us with broad knowledge of the state of the art regarding dependability and fog computing. We could find the answers to RQs which might help us for future adoption of fog computing technology for critical applications considering dependability requirements. The analysis also helped us to identify a number of research challenges that attracted authors attentions. Additionally, we noticed a number of challenges that are not still the topic of research in the fog computing community. In this Section we will present the identified research gaps as guidance for future research directions. We do not claim this list to be exhaustive.

Fog computing is a paradigm with certain characteristics that make it significantly different from other computing paradigms. In particular, fog computing is a resource constrained, geo-distributed and heterogeneous computing paradigm that must provide real-time responses for a large number of end devices. It is intensive from the communication perspective, with high volumes of data that can be transmitted horizontally at the Fog layer or vertically across layers (either upwards or downwards), and also from the computing perspective. It also requires support to mobility of end nodes. Thus, a dependability solution for fog computing must be well suited for the aforementioned characteristics.

What we believe is missing in the literature is: 1) Dependability solutions suitable for resources constrained fog computing infrastructure, 2) Reintegration of fog components after fault recovery in distributed systems, 3) Safety and Security aspects of dependability and 4) Domain-specific requirements.

1) Dependability solutions for resource constrained fog: Proposed methods for dependability solutions in the literature are mainly focusing on redundancy methods. Traditional redundancy methods, providing active/passive replicas of network components to be used in case of failure will let resources to be consumed or, in the best case, remain in stand-by mode. But fog computing is a resource constrained technology and efficient use of resources is a very important aspect to consider for designing fog networks. Research on dependability solutions for fog computing should be reconsidered to become resource optimized, i.e. reduce resource consumption and removing redundant components when possible. A potentil solution is to use methods like natural redundancy for link failures. Other dependability solutions considering limitation in resources for node and application failures are still under discussions and need to be explored. More specifically, dependability methods without using backup resources like, splitting data in different distributed resources, node migration and self-organization should be considered for further studies.

2) Reintegration after fault recovery in distributed systems: The proposed methods have mostly aimed to provide fault-tolerance solutions by replacing the component exposed to threat with other available components. However, reintegration and re-synchronization of data, applications, links and nodes in a distributed system has not been well explored. This is a fundamental feature for management of redundancy in long-life systems and scaling up without excessive resource utilization.

3) Safety and security aspects of dependability: The security and safety solutions proposed in the literature are mainly discussing traditional methods applicable to IoT. Novel solutions are also focusing on lightweight methods of encryption and data confidentiality which are very demanding considering resource limitations in fog computing. However, critical systems demand integrity and availability in extreme circumstances. In this case, only systematic security solutions and safety standards have higher priority to be developed and designed.

4) Domain-specific requirements: To comprehend dependability requirements for a fog-based network, it is necessary to have a broad understanding of its application. Different applications demand different dependability attributes. Dependability attributes, threats and solutions considered in the literature are mostly discussed in general as explained in Section V. Having a comprehensive general fault management framework, for instance to allow reconfigurable fault tolerance, might be useful in order to combine and integrate some of the proposed dependability solutions. However, application domain requirements should be considered as an important aspect in a multi-dimensional fault management framework.

VII. CONCLUSIONS

This paper provides an overview of current research on fog computing related dependability. Following a structured selection process we ended up with 27 scientific studies, from 2010–2018. Leveraging a systematic classification method, we have identified the current status of the topic and contributions of researchers. We answered four research questions in detail and discussed the results of the study.

ACKNOWLEDGMENT

This research has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 764785, and also from the VINNOVA project 2018-02437.

REFERENCES

[1] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, ser. MCC '12, Helsinki, Finland: ACM, 2012, pp. 13–16.

[2] M. Mukherjee, R. Matam, L. Shu, L. Maglaras, M. A. Ferrag, N. Choudhury, and V. Kumar, "Security and privacy in fog computing: Challenges," *IEEE Access*, vol. 5, pp. 19 293–19 304, 2017.

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

[4] OpenFog Consortium Architecture Working Group, "OpenFog Reference Architecture for Fog Computing," *OpenFog*, no. February, pp. 1–162, 2017.

[5] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering.," in *EASE*, vol. 8, 2008, pp. 68–77.

[6] A. Avizienis, J. C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no. 1, pp. 11–33, Jan. 2004, ISSN: 1545-5971.

[7] J. P. Arenas, "RCMBnet: A distributed hardware and firmware support for software fault tolerance," PhD thesis, Universitat de les Illes Balears, 2007.

[8] B. Kitchenham, R. Pretorius, D. Budgen, O. P. Brereton, M. Turner, M. Niazi, and S. Linkman, "Systematic literature reviews in software engineering: A tertiary study," *Information and Software Technology*, vol. 52, no. 8, pp. 792–805, 2010, ISSN: 0950-5849.

[9] S. Abbaspour Asadollah, D. Sundmark, S. Eldh, H. Hansson, and W. Afzal, "10 years of research on debugging concurrent and multicore software: A systematic mapping study," *Software Quality Journal*, vol. 25, no. 1, pp. 49–82, Mar. 2017, ISSN: 1573-1367.

[10] D. Budgen and P. Brereton, "Performing systematic literature reviews in software engineering," in *Proceedings of the 28th International Conference on Software Engineering*, ser. ICSE '06, Shanghai, China: ACM, 2006, pp. 1051–1052.

[11] M. S. Elbamby, M. Bennis, and W. Saad, "Proactive edge computing in latency-constrained fog networks," in *2017 European Conference on Networks and Communications (EuCNC)*, Jun. 2017, pp. 1–6.

[12] M. S. Elbamby, M. Bennis, W. Saad, M. Latva-aho, and C. S. Hong, "Proactive edge computing in fog networks with latency and reliability guarantees," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, no. 1, p. 209, Aug. 2018.

[13] A. Aral and I. Brandic, "Quality of service channelling for latency sensitive edge applications," in *2017 IEEE International Conference on Edge Computing (EDGE)*, Jun. 2017, pp. 166–173.

[14] K. E. Benson, G. Wang, N. Venkatasubramanian, and Y. Kim, "Ride: A resilient IoT data exchange middleware leveraging SDN and edge cloud resources," in *2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI)*, Apr. 2018, pp. 72–83.

[15] H. P. Breivold and K. Sandström, "Internet of things for industrial automation – challenges and technical solutions," in *2015 IEEE International Conference on Data Science and Data Intensive Systems*, 2015, pp. 532–539.

[16] J. A. Cabrera, D. E. Lucani., and F. H. P. Fitzek, "On network coded distributed storage: How to repair in a fog of unreliable peers," in *2016 International Symposium on Wireless Communication Systems (ISWCS)*, Sep. 2016, pp. 188–193.

[17] E. Cau, M. Corici, P. Bellavista, L. Foschini, G. Carella, A. Edmonds, and T. M. Bohnert, "Efficient exploitation of mobile edge computing for virtualized 5G in epc architectures," in *2016 4th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud)*, Mar. 2016, pp. 100–109.

[18] X. Chen, X. Wen, L. Wang, and W. Jing, "A fault-tolerant data acquisition scheme with mds and dynamic clustering in energy internet," in *2018 IEEE International Conference on Energy Internet (ICEI)*, May 2018, pp. 175–180.

[19] N. Chervyakov, M. Babenko, A. Tchernykh, N. Kucherov, V. Miranda-López, and J. M. Cortes-Mendoza, "AR-RRNS: Configurable reliable distributed data storage systems for internet of things to ensure security," *Future Generation Computer Systems*, vol. 92, pp. 1080–1092, 2019.

[20] V. S. Dasari, M. Pouryazdan, and B. Kantarci, "Selective versus non-selective acquisition of crowd-solicited IoT data and its dependability," in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, May 2018, pp. 1–6.

[21] C. Huang and K. Xu, "Reliable realtime streaming in vehicular cloud-fog computing networks," in *2016 IEEE/CIC International Conference on Communications in China (ICCC)*, Jul. 2016, pp. 1–6.

[22] M. Itani, S. Sharafeddine, and I. ElKabani, "Dynamic multiple node failure recovery in distributed storage systems," *Ad Hoc Networks*, vol. 72, pp. 1–13, 2018.

[23] A. Jonathan, M. Uluyol, A. Chandra, and J. Weissman, "Ensuring reliability in geo-distributed edge cloud," in *2017 Resilience Week (RWS)*, Sep. 2017, pp. 127–132.

[24] P. Kumar, G. Raj, and A. K. Rai, "A novel high adaptive fault tolerance model in real time cloud computing," in *2014 5th International Conference - Confluence The Next Generation Information Technology Summit (Confluence)*, Sep. 2014, pp. 138–143.

[25] J. Liu and Q. Zhang, "Offloading schemes in mobile edge computing for Ultra-Reliable Low Latency Communications," *IEEE Access*, vol. 6, pp. 12 825–12 837, 2018.

[26] R. Mennes, B. Spinnewyn, S. Latré, and J. F. Botero, "GRECO: A distributed genetic algorithm for reliable application placement in hybrid clouds," in *2016 5th IEEE International Conference on Cloud Networking (Cloudnet)*, Oct. 2016, pp. 14–20.

[27] K. C. Okafor, I. E. Achumba, G. A. Chukwudebe, and G. C. Ononiwu, "Leveraging fog computing for scalable IoT datacenter using spine-leaf network topology," *Journal of Electrical and Computer Engineering*, vol. 2017, 2017.

[28] O. Osanaiye, S. Chen, Z. Yan, R. Lu, K. R. Choo, and M. Dlodlo, "From cloud to fog computing: A review and a conceptual live vm migration framework," *IEEE Access*, vol. 5, pp. 8284–8300, 2017.

[29] F. Popentiu-Vladicescu and G. Albeanu, "Software reliability in the fog computing," in *2017 International Conference on Innovations in Electrical Engineering and Computational Technologies (ICIEECT)*, Apr. 2017, pp. 1–4.

[30] B. P. Rimal, D. Pham Van, and M. Maier, "Mobile-edge computing versus centralized cloud computing over a converged FiWi access network," *IEEE Transactions on Network and Service Management*, vol. 14, no. 3, pp. 498–513, Sep. 2017.

[31] M. T. Saqib and M. A. Hamid, "Fogr: A highly reliable and intelligent computation offloading on the internet of things," in *2016 IEEE Region 10 Conference (TENCON)*, Nov. 2016, pp. 1039–1042.

[32] M. Smara, M. Aliouat, A.-S. K. Pathan, and Z. Aliouat, "Acceptance test for fault detection in component-based cloud computing and systems," *Future Generation Computer Systems*, vol. 70, pp. 74–93, 2017, ISSN: 0167-739X.

[33] S. K. Sood, "SNA based QoS and reliability in fog and cloud framework," *World Wide Web*, vol. 21, no. 6, pp. 1601–1616, Nov. 2018.

[34] B. Spinnewyn, J. F. Botero, and S. Latré, "Cost-effective replica management in fault-tolerant cloud environments," in *2017 13th International Conference on Network and Service Management (CNSM)*, Nov. 2017, pp. 1–9.

[35] T. Wiss and S. Forsström, "Feasibility and performance evaluation of SCTP for the industrial internet of things," in *IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society*, Oct. 2017, pp. 6101–6106.

[36] Y. Xiao, Z. Ren, H. Zhang, C. Chen, and C. Shi, "A novel task allocation for maximizing reliability considering fault-tolerant in VANET real time systems," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Oct. 2017, pp. 1–7.

[37] B. Zhou, A. V. Dastjerdi, R. N. Calheiros, S. N. Srirama, and R. Buyya, "Mcloud: A context-aware offloading framework for heterogeneous mobile cloud," *IEEE Transactions on Services Computing*, vol. 10, no. 5, pp. 797–810, Sep. 2017.

[38] B. Spinnewyn, B. Braem, and S. Latre, "Fault-tolerant application placement in heterogeneous cloud environments," in *2015 11th International Conference on Network and Service Management (CNSM)*, Nov. 2015, pp. 192–200.