Resource Management in a Peer to Peer Cloud Network for IoT

Javadpour, Amir; Wang, Guojun; Rezaei, Samira

Published in:
Wireless personal communications

DOI:
10.1007/s11277-020-07691-7

Publication date:
2020

Wireless Personal Communications (2020) 115:2471–2488
https://doi.org/10.1007/s11277-020-07691-7

Resource Management in a Peer to Peer Cloud Network for IoT

Amir Javadpour[1] · **Guojun Wang**[1] · **Samira Rezaei**[2]

Published online: 10 August 2020

Abstract

Software-Defined Internet of Things (SDIoT) is defined as merging heterogeneous objects in a form of interaction among physical and virtual entities. Large scale of data centers, heterogeneity issues and their interconnections have made the resource management a hard problem specially when there are different actors in cloud system with different needs. Resource management is a vital requirement to achieve robust networks specially with facing continuously increasing amount of heterogeneous resources and devices to the network. The goal of this paper is reviews to address IoT resource management issues in cloud computing services. We discuss the bottlenecks of cloud networks for IoT services such as mobility. We review Fog computing in IoT services to solve some of these issues. It provides a comprehensive literature review of around one hundred studies on resource management in Peer to Peer Cloud Networks and IoT. It is very important to find a robust design to efficiently manage and provision requests and available resources. We also reviewed different search methodologies to help clients find proper resources to answer their needs.

Keywords Software-Defined Resource management · SDIoT Internet of Things (IoT) · Peer to Peer Fog Computing · Cloud Big Data · Bottlenecks of cloud networks

1 Introduction

Internet of Things (IoT) and Big Data is a new technology in which communications are beyond human to human, humans to machines and human to computers [1–3]. Billions or trillions of objects communicate with IoT and Big Data resources. The importance of IoT

✉ Guojun Wang
csgjwang@gzhu.edu.cn

Amir Javadpour
a_javadpour@e.gzhu.edu.cn

Samira Rezaei
s.rezaei.badafshani@rug.nl

1 School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou, China 510006

2 Bernoulli Institute for Mathematics and Computer Science, University of Groningen, Groningen, The Netherlands

has increased in several fields such as of industrial and transportation applications, various personal purposes such as electronic health care, smart cities, and so on. IoT is a communication revolution between Smart objects and is able to observe, listen, think and communicate without human intervention. IoT provides a platform for generated data by sensors and various hardware devices to be processed by data analysis systems such as machine learning and provides intelligent systems [4–8].

Cloud computing and Big Data empowers user-centric access and network-centric demand for a group of shared resources and services such as servers, storage space, repositories and application services. The aims of resource detection algorithms are to shorten task completion; increase system's operational capability and achieve maximum efficiency using existing computing resources. Process of resource detection for cloud computing systems includes following steps:

- Find and explore resources
- Allocate resource to desired work
- Perform tasks and return results
- Release resource

Big Data, Cloud Computing and IoT are based on two very dissimilar technologies which affect our lives. Cloud computing provides a convenient and consistent platform to access customizable computational resources and services have consistent interaction with service providers. The relationship between networks illustrated (Fig. 1). Cloud computing Networks architecture is divided into several parts: data center, infrastructure, design and application. Every one of these sections are considered as services to the upper layers and users of the lower sections. Cloud Networks are categorized into three major classes: software as a service (SaaS), platform as a service (PaaS) and Infrastructure as a service (IaaS) [9, 10]. SaaS delivers programs that run in the cloud environment. These programs are somehow accessible through small environments or network browsers. PaaS services provide a platform resources which users are able

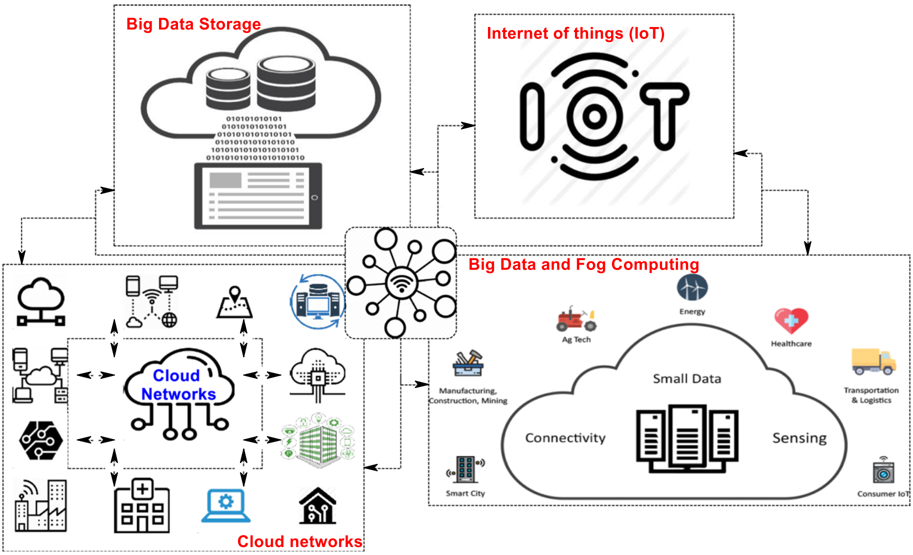

Fig. 1 Relationship between different networks

to use them on the provided platform. IaaS provides virtualized processing and computing resources over the internet [6, 7, 11–14]. Cloud computing and IoT have witnessed frequent and independent transformations. While they are very different, their characteristics are often complementary. IoT platform is able to use the available resources and capabilities of cloud computing instead of providing storage resources, processing and communication resources [6, 7, 15]. Cloud computing platform is capable of creating an transitional layer between objects and applications that hide all complex and necessary functional features [16–18]. In a scalable and completely heterogeneous scope, searching, locating, resources detection and services are very important. The concept of Cloud of Things (CoT) as a mixture of cloud network has been emerged to handle huge amount of Big Data [6, 7, 19, 20]. A challenges of CoT is to eliminate unnecessary communications between data. This will decrease avoidable loads in datacenters and causes unreliable delay. In addition, mobility (which is very important in IoT and Big Data) is not adequately supported in cloud computing. One solution is to use Fog-Computing and pre-process input data before transmitting them. Fog computing is a complementary model to cloud computing proposed by Cisco. The aggregation of Fog computing and IoT will lead to the creation of Fog of Things. It analyses sensitive data on the same source or transmitter device close to collection site to prevent sending large amounts of data to network. This affects the latency reduction in data analysis and risks in system security [6, 7, 21–24]. Fog computing is also helpful to control mobility issues by delivering resources and services for end users.

To make a CoT work, there are several items such as communication, storage and calculations. Sharing data and applications are very important in CoT and are in the category of communications. Cloud computing provides an efficient and inexpensive solution for communicating, tracking and managing any requests from any location using customized portals [21]. Although cloud computing substantially improves and facilitate IoT connections, it can still provide sensitive cases for some situations. Generated data in IoT has three important attributes: volume and quantity, variety and variety in terms of data type and velocity or frequency of data generation [25]. Moreover, devices which are typically used in IoT have limited processing power and energy resources which makes it almost impossible for them to do complicated and internal processing. This is why the collected data usually will be transmitted to powerful nodes in cloud computing resources to do all the calculations timely [26]. Cloud computing is also successful in expanding the boundaries of IoT to several networks with billions of devices [7, 27].

1.1 Introduction to IoT and It's Architecture

In 1999, the concept of IoT and interconnected objects with unique identifiers were introduced by Kevin Ashton. IoT is based on several other technologies such as wireless sensor networks (WSNs) [28, 29], barcodes, smart metering, NFC (Near-field communication), RFID (Radio-frequency identification), low power wireless communications, cloud computing, etc. which all play a key role. IoT defines a new generation of the Internet in which real objects are evaluated and identified. They can communicate and exchange information with another automatically (Fig. 2). Every object in IoT should be able to expand the information via Internet. In fact, IoT emphasizes interactions between networked objects [30, 31]. IoT ecosystem is important in many business, industrial, educational and health applications [32].Two architectures are presented for IoT ecosystem [33]:

- 3-layer architecture. The three layers in this architecture are called as (Network, perception App or application). The responsibility of perception layer is to identify

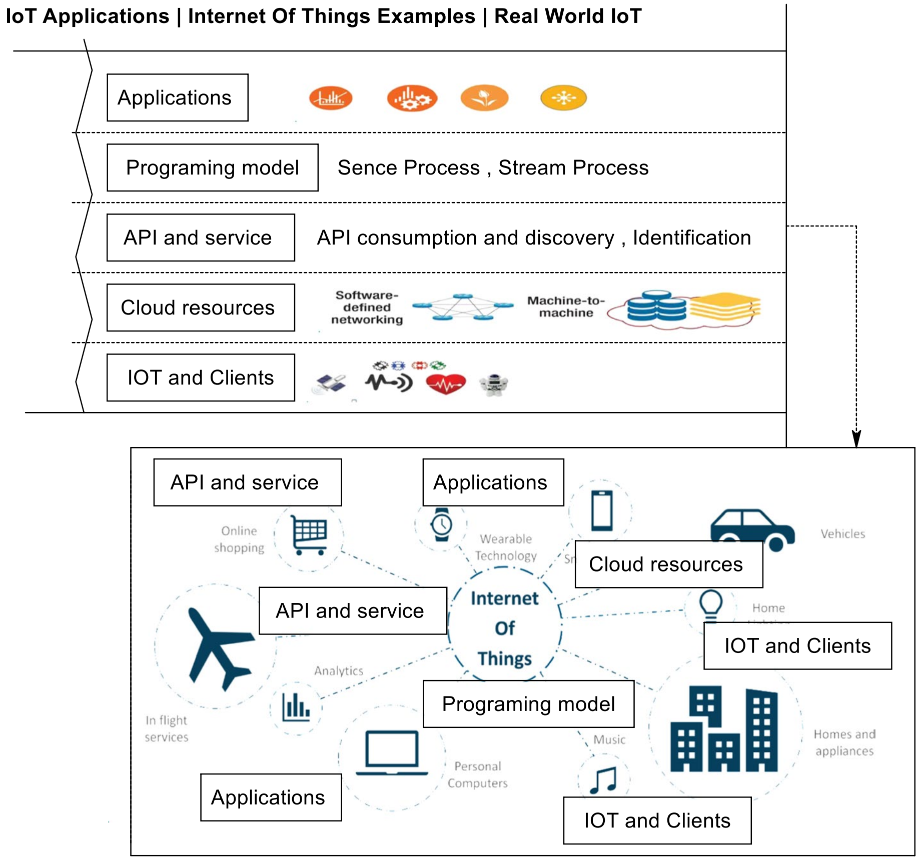

Fig. 2 Architecture IoT and layers

connected objects in the platform. This layer includes RFID tags, sensors, cameras, and etc. Network layer is the main layer and the core of the system. It transmits collected information by the previous layer and with the included hardware and software infrastructure, information and management centres. Finally, application layer aims to provide a platform which all the human interactions are covered with it.

- 5-layer architecture: 3-layer model was not suitable due to the expected development of IoT, so this architecture has been suggested. First layer is called Business and aims to manage and charge IoT applications and protect users' privacy. The application layer is intended to control applications used in IoT. It also develops more applications to enhance intelligence, authentication and safety of the platform. The processing layer which is sited on the third layer is responsible for handling the collected information in perception layer. The fourth layer is transport which transmits information between perception and processing layer as the network layer in a 3-layer architecture and includes communicative technologies like infrared, Wi-Fi, Bluetooth, etc. Each object in this layer is addressed via IPv6. The last layer is called perception aims to monitor and collect all the information about the physical characteristics of any connected objects to the system, such as position and temperature.

IoT is a large-scale distributed system with high heterogeneity hardware and software components [34]. Moreover, operation context of it is very dynamic and has a wide range of meanings, from software components to the geographic location of users which might be very mobile and animated. The key to determine some data should be processed according to connections given to one field is a context-awareness platform [6, 7, 35]. In IoT ecosystem, existing devices include limited resources, rich resources, passes, edge nodes, and cloud data centres [34]. As in the management process and resource allocation, IoT's mobile nature, the likelihood of the complexity of the service model, the current position and availability of connected devices and real-time processing in IoT should be considered, so this process is relative to traditional distributed systems and is more complicated than traditional cloud computing [36]. Figure 3 illustrates core of resource management activities with a hypothetical model for IoT ecosystem. It contains smart devices and intelligent objects in IoT; cloud and Fog nodes. One of the issues in resource detection is how to define resources in IoT ecosystem. Resource model should correctly represent various elements in the different rows of an IoT ecosystem, both the physical and virtual resources are also needed to be considered [4]. Each layer in IoT ecosystem has its own modelling requirements with different formats and languages [37, 38]. Or using Network Description Language to express existing descriptions [39].

1.2 Introduction to Resource Detection in IoT and Big Data

Typical resource detection process has two main phases: first, identifying and locating physical and local devices and then finding the resources or services offered by

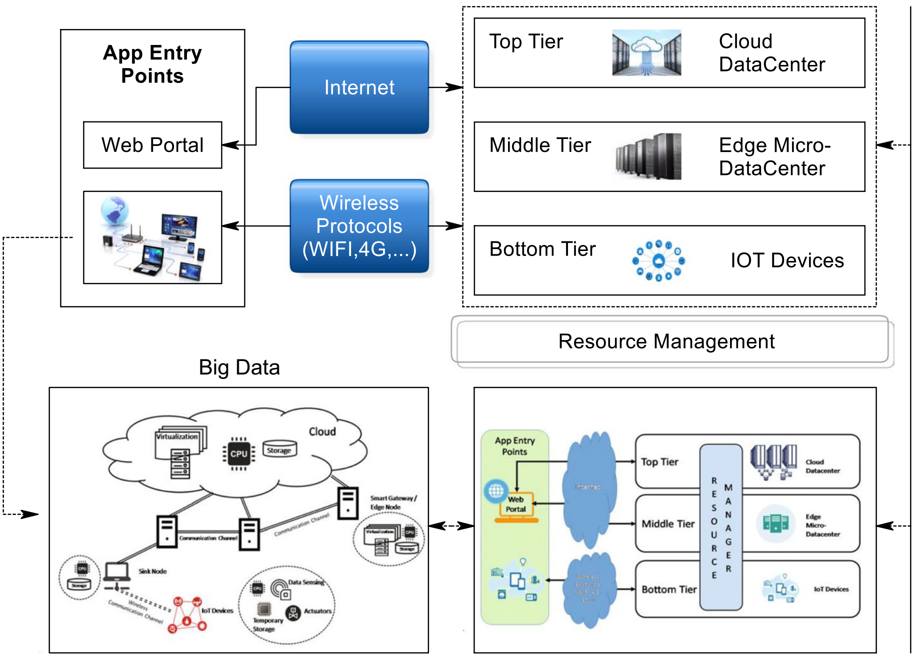

Fig. 3 Hypothetical model for IoT ecosystem

the device. IoT applications are not typically interested to access to a particular node in the network but the provided resources by that node are principal factors. IoT is well-suited to the concept of content networks in which identification and access to resources should preferably be based on the testimonials or content of resources. This is not the Internet works as it is based on IP and URI mapping. In addition to IPv6 that comes with the emergence of IoT, there are other designs for identifying devices and sensors [17, 40] such as semantic methods in which the addressing process is based on a semantic query on source or services description. The mapping between the actual address of devices and the provided description is transparent to the user or application.

Some authors like [41] have claimed that the performance of IoT and Big Data applications and IoT devices with each other as well as with information systems. Accepting a Service Oriented Architecture (SOA) approach naturally helps integrate devices with enterprise systems. Traditional SOA considered three roles that interact directly: service provider, service consumer, and a registry. Additionally, any SOA-based has some important functions: (discovery) (combination), and (access) [42–44]. Hlydra [45] is a smart link middleware that includes a resource-limited resource discovery mechanism. Authors in [46] specifically consider IoT scale and mobility, and proposes MobIoT as an IoT service to expresses a new discovery protocol. MobIoT takes care of registry performance and search functionality [47]. In [48] a resource detection infrastructure has been developed based on Q-learning techniques to enable context awareness. In this method, heterogeneous neighboring detection protocol are developed which reduce energy dissipation and detection delay. Infrastructure goal is to optimize above factors by learning a policy for mobile and static nodes. Presented method in [49] uses various sensor search techniques in terms of search quality, and demonstrates how its methodology improves QoS requirements. Article [50] emphasizes on a classification for detection technologies in IoT by defining an interactive detection pattern. These categories include search around me or around a specific client and search on the network. Figure 4 illustrates this technology.

It is stated in [51] that detection approaches are divided into local and remote scenarios. In local domain, detection takes place in a smart home environment on a local network. On the other hand, remote aspects work on smart city and global network. Detection methods are categorized as follows:

1. Distributed detection methods based on peer to peer coverage.

The philosophy behind such systems in p2p is using distributed hash table which supports multi-attribute queries in several domains. Authors in [52] developed Distributed Source Discovery architecture to communicate with each other through p2p protocol [53–55].

2. Centralized architecture to detect resources [56].
3. Constrained Application Protocol

COAP is application layer which includes a discovery mechanism and COAP has a web service on the kernel, which can respond to clients [57].

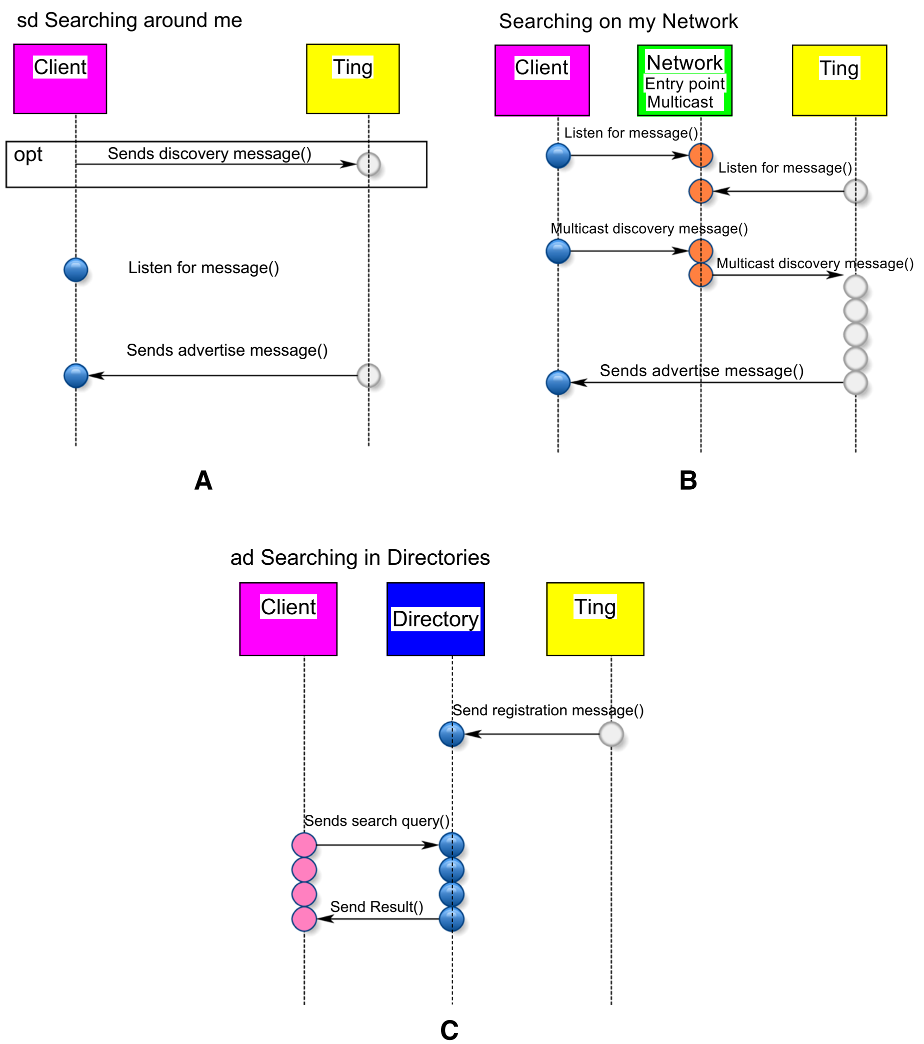

Fig. 4 Iinteractive pattern for different search methodologies: Around me, Search on my network and Search in directory

1.3 A Comparative Study of Resource Detection in Computer Networks

Big Data,Grid computing and cloud computing systems provide large number of computing resources and services to their users. Resource detection for providing a collection of resources and attributes are expressed by their owners. Resources are able to satisfy a predefined set of users' requirements. There are five main resource detection mechanisms classes: 1-(centralized or intensive), 2-(decentralized), 3-(peer to peer network), 4-(Hierarchical structure), and 5-(Agent-based modelling). Some factors such as high number of resources, distribution ownership, resource heterogeneity and spoilage, dynamism, and resource deviations make it challenging to detect and assign resources for user's requests

[24, 58]. In centralized mechanisms, a special collection of controllers detect resources-required which match with the client server architecture. In this scenario, servers collect information about the available services [59]. Figure 5 summarises the differences between existing mechanisms in centralized category along with their disadvantages and advantages. On the other hand, centralized mechanisms for resource detection are inadequate for large-scale computer networks and researchers have studied decentralized techniques to solve this issue. In these mechanisms, central databases or servers are eliminated and all nodes work together to perform resource detection on large-scale systems. Although these mechanisms work better on large-scale networks, they create overloads when managing network architecture. Figure 6 represents some popular and well-known mechanisms in this category.

The next model of resource detection is hierarchical mechanism in which resources information are modernized under a series of indexed nodes. The distribution system is created by categorizing resources in different clusters. In these methods, The queries are follow out based on hierarchically because the servers are organized in a hierarchical manner. Each server is responsible for partitioning resources. In Fig. 7 some presented methods in this mechanism have been reviewed [59]. Figure 8 reviews the advantages and disadvantages of some presented methods in hierarchical mechanism.

LARD [60] is a distributed resource detection technique using learning automata to find the shortest route between users and peer. This method supports multithreading queries. IAPS [61] is based on ant optimization files in network. For each type of file,

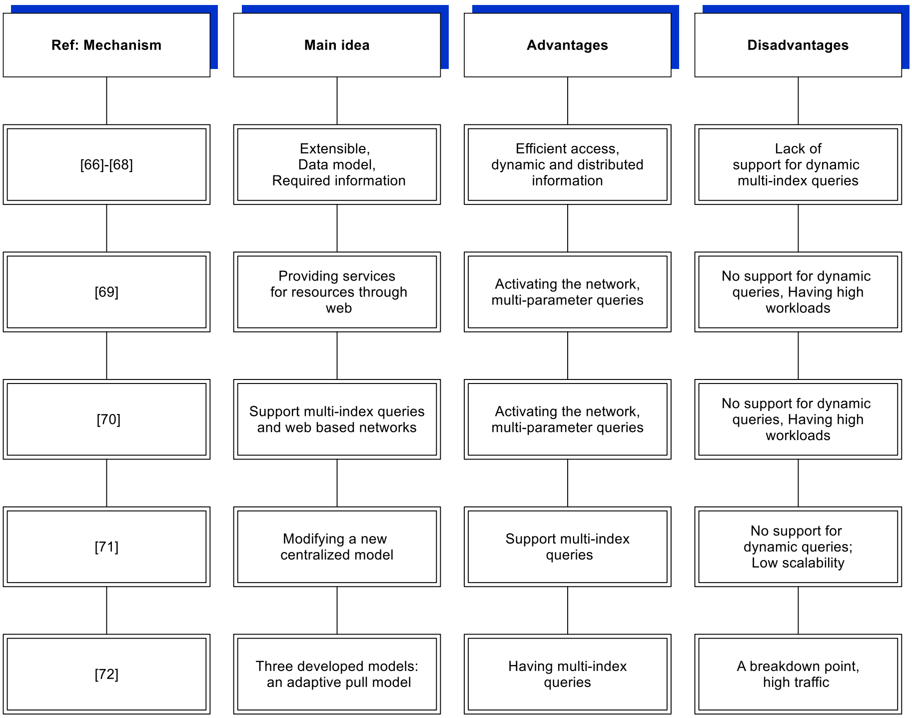

Fig. 5 A comparison centralised mechanisms

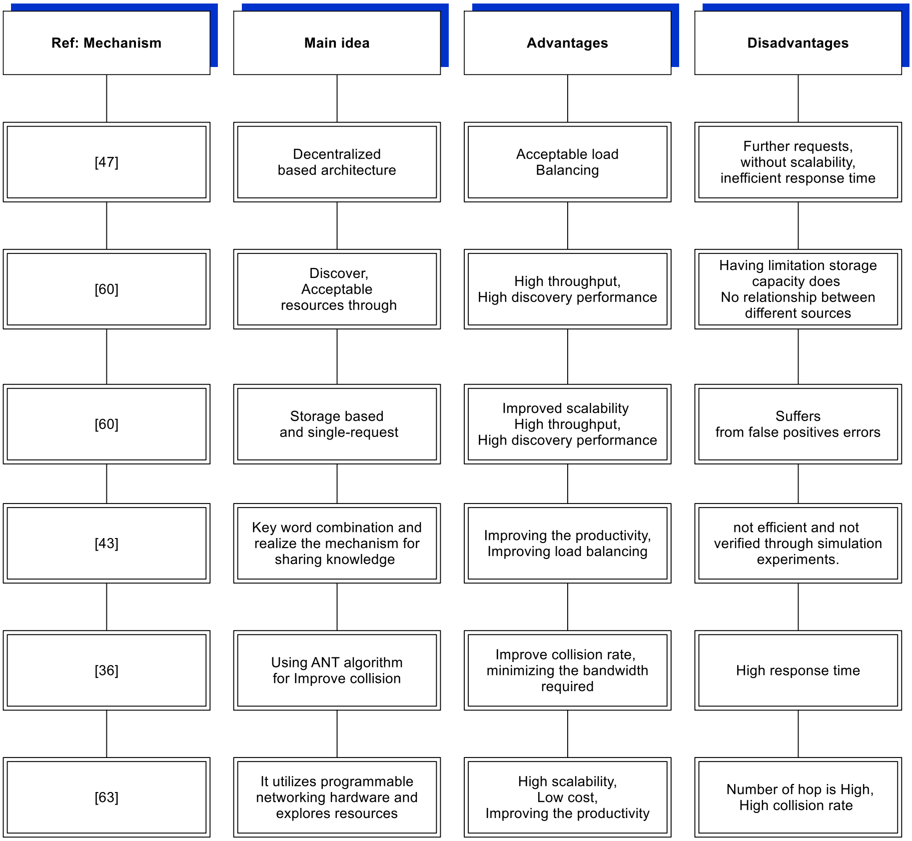

Fig. 6 Decentralized mechanisms

it considers a score based on previous searches, to minimize search area and overload. DHMCF [62] is based on the state model with high respond rate to dynamic requests. Each machine measures its state based on arrived requests and the state model of network. This model estimates the state of machines before and after accepting requests; therefore, machines are aware of their status to decide whether to accept or refuse a request. Most techniques in this category suffer from a sudden change in broad network coverage and low security and are not scalable. In contrast, they support dynamic and multi-attribute queries.

Structured peer-to-peer networks are not flexible. Important information is stored in a specific peer which uses distributed hash table for direct search through the data. In these networks, each peer manages a subspace to stores information about connected peers. A well-known example is Chord [18] in which distributed hash table stores key-value pairs. Chord is not suitable for dynamic and multi-attribute queries, but it supports load balancing. CAN is another model for structured peer-to-peer network [63]. Like Chord, it uses more than hexing subordinate in distributed hash table to support repercussion. Other methods such as D2B [64] and SCAN [65] have similar functionality to CAN in peer-to-peer networks.

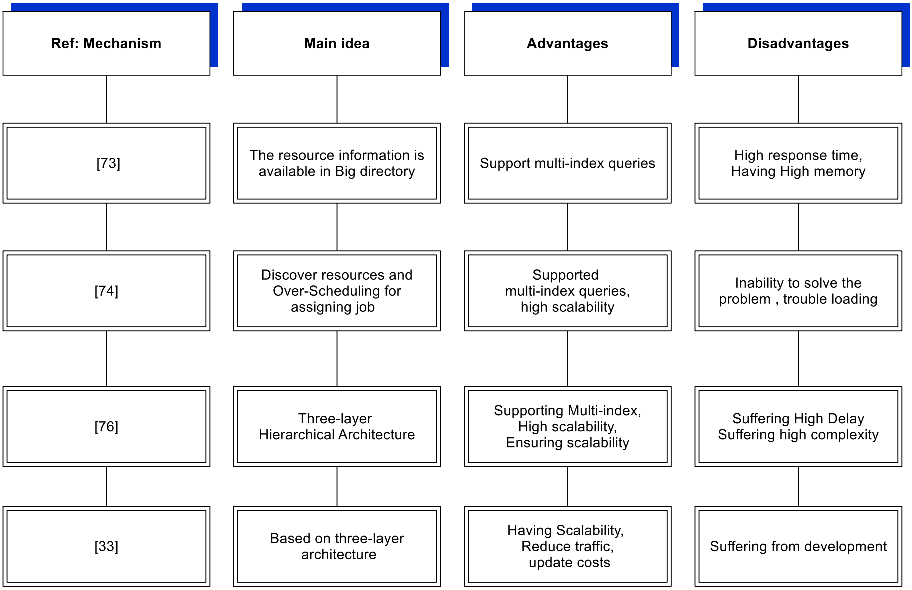

Fig. 7 Hierarchical mechanisms

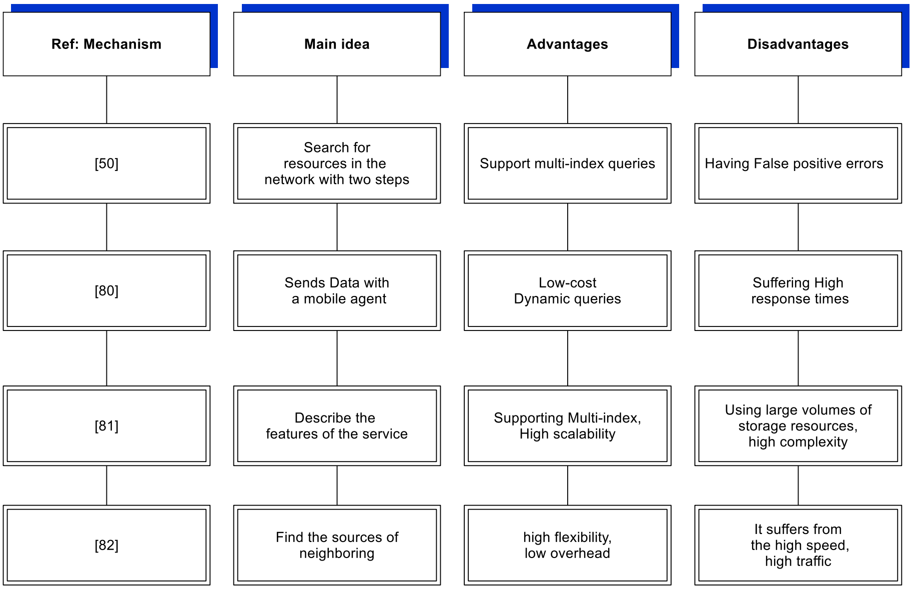

Fig. 8 Agent-based mechanisms

Paper [44] proposed a one-dimensional decentralized technique based on DHT in which each source has multiple attributes. This study uses a different strategy than CAN and Chord to map data records to index stead. Local data elements share same prefixes in words. It is scalable, flexible and benefits from load balancing. On the other hand, it does not support error tolerance, security and geographic location. Other similar approaches are presented in [66].

Super-P2P networks are used to facilitate convergence of peer-to-peer networks. A few peers work as super-peers to manage assigned other peers in networks [42, 67]. Super peers have special resources (memory, computing, and bandwidth). Peers need to be assigned to at-least one super. Supers are responsible for routing all requests [68]. Super peers form a structure similar to a typical p2p network and are mainly computational and communication centers. Semi-centralized search is one of the advantages of this asymmetric model. KaZaA conforms Super-peer model, in which there are two distinct types of privileged and ordinary peers. Super peers handle a set of peers. Ordinary peers update their super peer about their resource indexes. This helps to have better load balancing, less search time, more scalability [67]. In Gnutella2, super-nodes search their data tables locally and connect each request to corresponding peer [69]. If the answer is not accessible locally, the request will be sent to other super nodes. This technique greatly reduces network traffic and increases reliability. HPRDG [67] detects resource in cubic computational grids using super-peers. HPRDG connects two layers of SN and Chyper SN with ring topology.

Hybrid networks are presented to overcome the disadvantages of structured and unstructured topologies. In [63], structured search techniques, such as DHT, can index and locate rare items and to use overturning techniques to locate repetitive items. HybridFlood [70] is another hybrid model using flooding and super-peers. Algorithm allows to use flooding with limited number of steps in the first phase. In second phase, it finds peers with maximum number of links to other peers. This method improves search performance by reducing additional messages per step.

1.4 Resource Detection in Cloud Computing and Big Data

Different approaches have been proposed to detect resource in cloud computing and Big Data and SDN (Software-defined networking). Some articles like [71–73] uses history of resource detection and transitory clustering services. This mechanism creates clusters of grouped nodes over time. It is suitable for large, heterogeneous, and dynamic cloud computing environments in terms of flexibility, scalability, high tolerance. On the other hand, it does not have proper monitoring of the local interface. Researchers in [56] presents a method which uses limited, local, multi-attributed search. In this method, all data and queries are stored decentralized within all physical machines in data centers. Figure 9 summarizes the advantages and disadvantages of some resource detection mechanisms in cloud computing.

1.5 Resource Detection in IoT and Fog computing

Resource allocation in a system requires search and detection of available resources [74, 75]. Following requirements should also be noted in resource detection [76]:

1.5.1 Flexible Identification Scheme

Internet which unique IP and URLs are enough to identify single node across web; IoT systems still need proper mechanisms to detect resources. The purpose of the resource

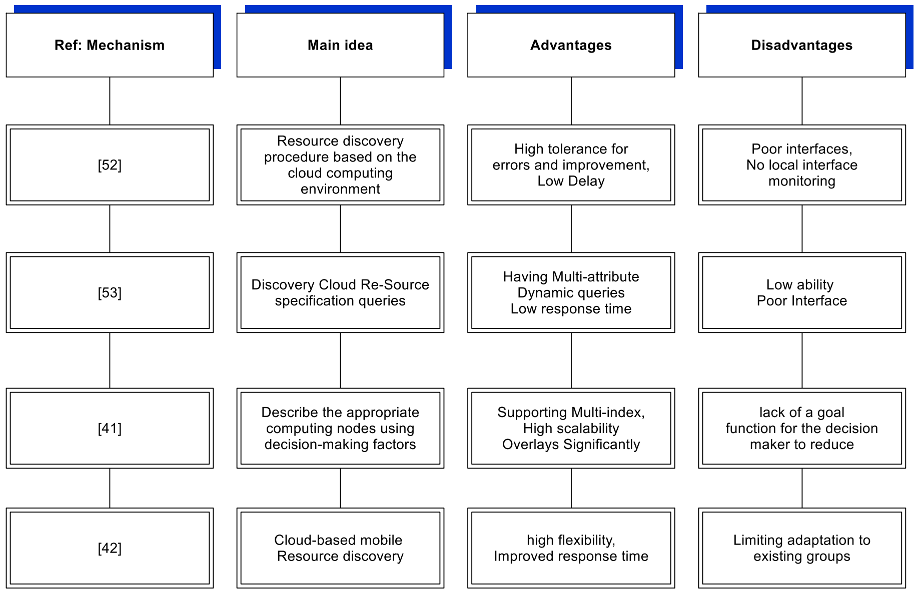

Fig. 9 Source discovery mechanisms in cloud computing

detection on IoT is not the device itself, but the sources that it produces. No standard has been currently developed for this type of ID. Some identifiers used in IoT applications are EPC (Electronic-Product-Code), UPC (Global-Product-Code), in addition to URL and IPv6.

1.5.2 Support for Multi attribute and Rang Query

Detection mechanism must have the ability to attaint (queries) through the exact and correct matching Given ID; it also able to take-queries with more attributes (such as Place and category). Additionally, along with correct matching of queries, the detection The system must support a query that specifies upper and lower defined sill one or more attributes [77].

1.5.3 Context Awareness

Context-awareness can used to describe the status of physical entities like automaton and application [78]. Context involves location, identity, status of individuals, groups, and computing objects in relation to provided data by devices. It allows IoT to offer resources based with minimum human interaction and facilitates data interpretation and communications. Recently, several articles for source detection have emerged in IoT, though none of them completely covers all of the above requirements. In the next section, we will examine some of them.

1.6 Articles About Resource Detection

Article [4] describes the distributed infrastructure design, which aims at allowing smart things to communicate and collaborate with the consideration of space repartition and their wide criterion. The proposed foundation has several features: Awareness, Self-management and User-friendly. Awareness is achieved by storing site Info to Global routing may be prevented. The method often enables context-awareness services. Self-management means detection of shrewd harness by subtraction, as well as shape and renovation. The developed interface is easy to understand for human users and machines to rummage for sets presented by intelligent harness. On ascendancy this arrangement-architecture there is a Scalability mechanism that solves on resources. Such a thing automaton provides location awareness as well as load-balancing because they are automatically routed-to-destination that are far away from overloaded infrastructure.

2 Conclusion

Resource management as a process of allocating different types of resources such as Fog computing, Big Data Infrastructure, IoT networking and energy resources to a set of requests from different clients is becoming a major concept to consider in cloud environments. The allocated resources should meet several criteria such as performance objectives, infrastructure providers and users need. There are several methods to detect available resources in the network. We categorized existing studies in the literature into centralized, decentralized, hierarchical and agent-based mechanisms. Each one of these technologies has advantages and disadvantages which should be considered when a new client asks for resources. The provided solution must consider application performance, resource availability and cost-effective scaling of available resources. Dynamic changes of applications demand should also be considered. This paper reviews a broad ranges of presented resource management techniques in the literatures and provides a classification of methods with their advantages and disadvantages compare to other available methodologies.

Acknowledgements This work is supported in part by the National Natural Science Foundation of China under Grants 61632009 & 61472451, in part by the Guangdong Provincial Natural Science Foundation under Grant 2017A030308006 and High-Level Talents Program of Higher Education in Guangdong Province under Grant 2016ZJ01.

References

1. Li, Y., Cheng, X., Cao, Y., Wang, D., & Yang, L. (2018). Smart choice for the smart grid: narrowband internet of things (NB-IoT). *IEEE Internet of Things Journal, 5*(3), 1505–1515.
2. Ammar, M., Russello, G., & Crispo, B. (2018). Internet of things: A survey on the security of IoT frameworks. *Journal of Information Security and Applications, 38,* 8–27.
3. Javadpour, A., Wang, G., Rezaei, S., & Li, K.-C. (2020). Detecting straggler MapReduce tasks in big data processing infrastructure by neural network.*The Journal of Supercomputing*.
4. Xu, L. D., He, W., & Li, S. (2014). Internet of Things in Industries: A Survey. *IEEE Transactions on Industrial Informatics, 10*(4), 2233–2243.
5. Park, E., Del Pobil, A. P., & Kwon, S. J. (2018). The role of internet of things (IoT) in smart cities: Technology roadmap-oriented approaches. *Sustainability, 10*(5), 1388.
6. Javadpour, A., Saedifar, K., Wang, G., & Li, K.-C. (2020). Optimal execution strategy for large orders in big data: Order type using q-learning considerations. *Wireless Personal Communications*.

7. Javadpour, A., Kazemi Abharian, S., & Wang, G. (2017). Feature selection and intrusion detection in cloud environment based on machine learning algorithms, In *2017 IEEE International Symposium* on *Parallel* and *Distributed Processing with Applications 2017 IEEE International Conference on Ubiquitous Computing* and *Communications* (pp. 1417–1421).
8. Mohammadi, A., & Professor, A. (2016). Improving brain magnetic resonance image (MRI) segmentation via a novel algorithm based on genetic and regional growth. *Journal of Biomedical Physics and Engineering, 6*(2), 95–108.
9. Pflanzner, T., & Kertesz, A. (2016). A survey of IoT cloud providers, In *2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 730–735).
10. Renner, T., Kliem, A., & Kao, O. (2014). The device cloud—Applying cloud computing concepts to the internet of things. In *2014 IEEE 11th Intl Conf on Ubiquitous Intelligence and Computing and 2014 IEEE 11th Intl Conf on Autonomic and Trusted Computing and 2014 IEEE 14th Intl Conf on Scalable Computing and Communications and Its Associated Workshops*, (pp. 396–401).
11. Bitam, S., Mellouk, A., & Zeadally, S. (2015). VANET-cloud: a generic cloud computing model for vehicular Ad Hoc networks. *IEEE Wireless Communications, 22*(1), 96–102.
12. Xhafa, F., Barolli, L., & Amato, F. (2016). *Advances on P2P, Parallel, Grid, Cloud and Internet Computing: Proceedings of the 11th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC–2016) November 5–7, 2016, Soonchunhyang University, Asan, Korea*. Springer International Publishing.
13. He, J., Zhang, Y., Lu, J., Wu, M., & Huang, F. (2018). Block-stream as a service: A more secure, nimble, and dynamically balanced cloud service model for ambient computing. *IEEE Network, 32*(1), 126–132.
14. Mirmohseni, S. M., Tang, C., & Javadpour, A. (2020). Using markov learning utilization model for resource allocation in cloud of thing network. *Wireless Personal Communications*.
15. Iamnitchi, A., & Foster, I. (2004). A Peer-to-Peer Approach to resource location in grid environments. In J. Nabrzyski, J. M. Schopf, & J. Węglarz (Eds.), *Grid resource management: State of the art and future trends* (pp. 413–429). Boston: Springer.
16. Yannuzzi, M., Milito, R., Serral-Gracià, R., Montero, D., & Nemirovsky, M. (2014). Key ingredients in an IoT recipe: Fog computing, cloud computing, and more fog computing. In *2014 IEEE 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)* (pp. 325–329).
17. Jara, A. J., Lopez, P., Fernandez, D., Castillo, J. F., Zamora, M. A., & Skarmeta, A. F. (2013). Mobile Digcovery: A global service discovery for the internet of things. In *2013 27th International Conference on Advanced Information Networking and Applications Workshops*, (pp. 1325–1330).
18. Uehara, M. (2015). A case study on developing cloud of things devices. In *2015 Ninth International Conference on Complex, Intelligent, and Software Intensive Systems*, (pp. 44–49).
19. Prazeres, C., & Serrano, M. (2016). SOFT-IoT: Self-organizing FOG of things. In *2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA)* (pp. 803–808).
20. Chen, S., Zhang, T., & Shi, W. (2017). Fog computing. *IEEE Internet Computing, 21*(2), 4–6.
21. Kaur, D., & JyotsnaSengupta, G. (2007). Resource discovery in web-services based grids. *World Academy of Science, Engineering and Technology, 31,* 284–288.
22. Singh, D., Tripathi, G., & Jara, A. J. (2014). A survey of Internet-of-Things: Future vision, architecture, challenges and services. In *2014 IEEE World Forum on Internet of Things (WF-IoT)* (pp. 287–292).
23. Liu, W., Nishio, T., Shinkuma, R., & Takahashi, T. (2014). Adaptive resource discovery in mobile cloud computing. *Computer and Communications, 50,* 119–129.
24. Navimipour, N. J., Rahmani, A. M., Navin, A. H., & Hosseinzadeh, M. (2014). Resource discovery mechanisms in grid systems: A survey. *Journal of Network and Computer Applications, 41,* 389–410.
25. Gia, T. N., Jiang, M., Rahmani, A., Westerlund, T., Liljeberg, P., & Tenhunen, H. (2015). Fog Computing in Healthcare Internet of Things: A Case Study on ECG Feature Extraction. In *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing* (pp. 356–363).
26. Khodadadi, F., Dastjerdi, A. V., & Buyya, R. (2015). Simurgh: A framework for effective discovery, programming, and integration of services exposed in IoT. In *2015 International Conference on Recent Advances in Internet of Things (RIoT)* (pp. 1–6).

27. Bessis, N., & Dobre, C. (2014). *Big data and internet of things: A roadmap for smart environments*. Berlin: Springer.
28. Javadpour, A., & Memarzadeh-Tehran, H. (2015). A wearable medical sensor for provisional healthcare, In *2015 2nd International Symposium on Physics and Technology of Sensors (ISPTS)* (pp. 293–296).
29. Javadpour, A., Memarzadeh-Tehran, H., & Saghafi, F. (2015). A temperature monitoring system incorporating an array of precision wireless thermometers. In *2015 International Conference on Smart Sensors and Application (ICSSA)* (pp. 155–160).
30. Aazam, M., & Huh, E. (2014) Fog computing and smart gateway based communication for cloud of things. In *2014 International Conference on Future Internet of Things and Cloud* (pp. 464–470).
31. Lv, W., Meng, F., Zhang, C., Lv, Y., Cao, N., & Jiang, J. (2017). A general architecture of IoT system. In *2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)* (Vol. 1, pp. 659–664).
32. Kumar, N. (2013). Smart and intelligent energy efficient public illumination system with ubiquitous communication for smart city. In *International conference on smart structures and systems—ICSSS'13* (pp. 152–157).
33. Hachem, S., Pathak, A., & Issarny, V. (2014). Service-oriented middleware for large-scale mobile participatory sensing. *Pervasive and Mobile Computing, 10,* 66–82.
34. Chang, R.-S., & Hu, M.-S. (2010). A resource discovery tree using bitmap for grids. *Future Generation Computing Systems, 26*(1), 29–37.
35. Botta, A., de Donato, W., Persico, V., & Pescapé, A. (2016). Integration of Cloud computing and Internet of Things: A survey. *Future Generation Computing Systems, 56,* 684–700.
36. Compton, M., et al. (2012). The SSN ontology of the W3C semantic sensor network incubator group. *Journal of Web Semantics, 17,* 25–32.
37. Brocco, A., Malatras, A., & Hirsbrunner, B. (2010). Enabling Efficient Information Discovery in a Self-structured Grid. *Future Generation Computing Systems, 26*(6), 838–846.
38. Distefano, S., Merlino, G., & Puliafito, A. (2012). Enabling the cloud of things. In *Proceedings of the 2012 6th international conference on innovative mobile and internet services in ubiquitous computing* (pp. 858–863).
39. Eisenhauer, M., Rosengren, P., & Antolin, P. (2009). A development platform for integrating wireless devices and sensors into ambient intelligence systems. In *2009 6th IEEE annual communications society conference on sensor, mesh and ad hoc communications and networks workshops* (pp. 1–3).
40. Brogi, A., Popescu, R., Gutiérrez, F., López, P., & Pimentel, E. (2008). A service-oriented model for embedded peer-to-peer systems. *Electronic Notes in Theoretical Computer Science, 194*(4), 5–22.
41. Datta, S. K., Da Costa, R. P., & Bonnet, C. (2015). Resource discovery in internet of things: current trends and future standardization aspects. In *2015 IEEE 2nd world forum on internet of things (WF-IoT)(WF-IOT)* (Vol. 00, pp. 542–547).
42. Kalapriya, K., Nandy, S. K., Srinivasan, D., Uma Maheshwari, R., & Satish, V. (2004). A framework for resource discovery in pervasive computing for mobile aware task execution. In *Proceedings of the 1st conference on computing frontiers* (pp. 70–77).
43. Akherfi, K., Gerndt, M., & Harroud, H. (2018). Mobile cloud computing for computation offloading: Issues and challenges. *Applied Computing and Informatics, 14*(1), 1–16.
44. Li, H., & Liu, L. (2007). A decentralized resource discovery based on keywords combinations and node clusters in knowledge grid BT—Advanced intelligent computing theories and applications. With aspects of theoretical and methodological issues (pp. 738–747).
45. Yang, L. T., & Guo, M. (2005). *High-performance computing: Paradigm and infrastructure*. New York: Wiley.
46. Jennings, N. R. (2001). An agent-based approach for building complex software systems. *Communications of the ACM, 44*(4), 35–41.
47. Yan, L., Shen, H., & Chen, K. (2017). MobiT: A distributed and congestion-resilient trajectory based routing algorithm for vehicular delay tolerant networks. In *2017 IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI)* (pp. 209–214).
48. Huber, S., Seiger, R., Kühnert, A., & Schlegel, T. (2016). Using semantic queries to enable dynamic service invocation for processes in the internet of things. In *2016 IEEE Tenth International Conference on Semantic Computing (ICSC)* (pp. 214–221).
49. Manvi, S. S., & Shyam, G. K. (2014). Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey. *Journal of Network and Computer Applications, 41,* 424–440.
50. Alhakbani, N., Hassan, M. M., Hossain, M. A., & Alnuem, M. (2014). A framework of adaptive interaction support in cloud-based internet of things (IoT) environment. In *Internet and Distributed Computing Systems* (pp. 136–146).

51. Kakarontzas, G., & Savvas, I. K. (2006). Agent-based resource discovery and selection for dynamic grids. In *15th IEEE international workshops on enabling technologies: infrastructure for collaborative enterprises (WETICE'06)* (pp. 195–200).
52. Delicato, F. C., Pires, P. F., & Batista, T. (2017).The activities of resource discovery and resource estimation BT—Resource management for internet of things, Delicato, F. C., Pires, P. F., & Batista, T. (Eds). Cham: Springer (pp. 33–44).
53. Zarrin, J., Aguiar, R. L., & Barraca, J. P. (2018). Resource discovery for distributed computing systems: A comprehensive survey. *Journal of Parallel and Distributed Computing, 113,* 127–166.
54. Caraguay, Á. L. V., & Villalba, L. J. G. (2017). Monitoring and discovery for self-organized network management in virtualized and software defined networks. *Sensors, 17*(4), 731.
55. Sedaghat, M., Hernández-Rodríguez, F., & Elmroth, E. (2014). Autonomic resource allocation for cloud data centers: A Peer to Peer Approach. In *2014 International conference on cloud and autonomic computing* (pp. 131–140).
56. Armbrust, M., et al. (2010). A view of cloud computing. *Communications of the ACM, 53*(4), 50–58.
57. Manna, S., Bhunia, S. S., & Mukherjee, N. (2014). Vehicular pollution monitoring using IoT. In *International conference recent adv innovations engineering* (pp. 1–5).
58. Hasanzadeh, M., & Meybodi, M. R. (2013). Grid resource discovery based on distributed learning automata. *Computing, 96,* 909–922.
59. Kovvur, R. M. R., Kadappa, V., Ramachandram, S., & Govardhan, A. (2010). Adaptive resource discovery models and Resource Selection in grids. In *2010 first international conference on parallel, distributed and grid computing (PDGC 2010)* (pp. 95–100).
60. Toninelli, A., Corradi, A., & Montanari, R. (2008). Semantic-based discovery to support mobile context-aware service access. *Computer and Communications, 31*(5), 935–949.
61. Djamaa, B., & Yachir, A. (2016). A proactive trickle-based mechanism for discovering CoRE resource directories. *Procedia Computer Science, 83,* 115–122.
62. Mell, P. M., & Grance, T. (2011). SP 800-145. The NIST Definition of Cloud Computing. National Institute of Standards & Technology, Gaithersburg, MD, United States.
63. Whitmore, A., Agarwal, A., & Da Xu, L. (2015). The Internet of Things—A survey of topics and trends. *Information Systems Frontiers, 17*(2), 261–274.
64. Torkestani, J. A. (2013). A multi-attribute resource discovery algorithm for peer-to-peer grids. *Applied Artificial Intelligence, 27*(7), 575–598.
65. Razzaque, M. A., Milojevic-Jevric, M., Palade, A., & Clarke, S. (2016). Middleware for Internet of Things: A Survey. *IEEE Internet of Things Journal, 3*(1), 70–95.
66. Satyanarayanan, M., Bahl, P., Caceres, R., & Davies, N. (2009). The case for VM-based cloudlets in mobile computing. *IEEE Pervasive Computing, 8*(4), 14–23.
67. Narendra, K. S., & Thathachar, M. A. L. (2012). *Learning automata: An introduction*. Chelmsford: Courier Corporation.
68. Tan, Y. (2009) A Multi-agent Approach for P2P Based Resource Discovery in Grids. In *2009 International Joint Conference on Artificial Intelligence* (pp. 43–45).
69. Bitam, S., & Mellouk, A. (2012). ITS-cloud: Cloud computing for Intelligent transportation system. In *2012 IEEE Global Communications Conference (GLOBECOM)* (pp. 2054–2059).
70. Gao, G., Li, R., Wen, K., & Gu, X. (2012). Proactive replication for rare objects in unstructured peer-to-peer networks. *Journal of Network and Computer Applications, 35*(1), 85–96.
71. Alhakbani, N., Hassan, M. M., Hossain, M. A., & Alnuem, M. (2014). A Framework of adaptive interaction support in cloud-based internet of things (IoT) environment BT—Internet and distributed computing systems (pp. 136–146).
72. Javadpour, A. (2020). Providing a way to create balance between reliability and delays in SDN networks by using the appropriate placement of controllers. *Wireless Personal Communications*, *110*(2), 1057–1071.
73. Javadpour, A. (2019) Improving resources management in network virtualization by utilizing a software-based network. *Wireless Personal Communications*, *106*(2), 505-519

74. Elmroth, E., & Tordsson, J. (2005). An interoperable, standards-based grid resource broker and job submission service. In *First International Conference on e-Science and Grid Computing (e-Science'05)* (pp. 9–220).
75. Aggarwal, D. K., & Aron, R. (2017). IoT based Platform as a service for provisioning of concurrent applications, *CoRR*, abs/1711.1.
76. Hameurlain, A., Cokuslu, D., & Erciyes, K. (2010). Resource discovery in grid systems; a survey. *International Journal of Metadata, Semantics and Ontologies, 5*(3), 251–263.
77. Kang, J., & Sim, K. M. (2012). A multiagent brokering protocol for supporting Grid resource discovery. *Applied Intelligence, 37,* 527–542.
78. Butt, F., Bokhari, S. S., Abhari, A., & Ferworn, A. (2011). Scalable grid resource discovery through distributed search. *CoRR*, abs/1110.1.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Amir Javadpour received the BSc in ICT in 2010 from Tamishan University of Behshahr, Iran, and received his MSc degree in 2014 in Medical Information Technology Engineering from University of Tehran, Iran. He is currently a Ph.D. Candidate at Guangzhou University, China. His research interests include Cloud Computing, Steganography, Wireless Target Tracking, Wireless Body Area Network (WBANs), Big Data, Intrusion detection systems (IDS), Healthcare Real time monitoring measurement, Machine Learning, Wireless Sensor Network, SDN network and Secure Communication.

Guojun Wang received B.Sc. degree in Geophysics, M.Sc. degree in Computer Science, and Ph.D. degree in Computer Science, at Central South University, China, in 1992, 1996, 2002, respectively. He is a Pearl River Scholarship Distinguished Professor of Higher Education in Guangdong Province, a Doctoral Supervisor and Vice Dean of School of Computer Science and Cyber Engineering, Guangzhou University, China, and the Director of Institute of Computer Networks at Guangzhou University. He has been listed in Chinese Most Cited Researchers (Computer Science) by Elsevier in the past six consecutive years (2014-2019). His research interests include artificial intelligence, big data, cloud computing, Internet of Things (IoT), blockchain, trustworthy/dependable computing, network security, privacy preserving, recommendation systems, and smart cities. He is a Distinguished Member of CCF, a Member of IEEE, ACM and IEICE.

Samira Rezaei received her BSc and MSC in Information Technology. She graduated from University of Tehran, Iran on 2014 working on data science techniques. She is currently a Ph.D. Candidate at the University of Groningen, The Netherlands. Her current research involves applying data analysis techniques on astronomical datasets. Her main research interests are Big data, Deep learning, cloud computing, and the Internet of things.