

# A Comprehensive Survey on Resource Management in Internet of Things

Lokesh B. Bhjantri<sup>1</sup> and Gangadharaiyah S.<sup>2</sup>

<sup>1</sup> Department of Information Science and Engineering, Basaveshwar Engineering College, Bagalkot, India

<sup>2</sup> Department of Computer Science, S.R.S.M.N. Government First Grade College, Barkur, India

<https://doi.org/10.26636/jtit.2020.145220>

**Abstract**—Efficient resource management is a challenging task in distributed systems, such as the Internet of Things, fog, edge, and cloud computing. In this work, we present a broad overview of the Internet of Things ecosystem and of the challenges related to managing its resources. We also investigate the need for efficient resource management and the guidelines given/suggested by Standard Development Organizations. Additionally, this paper contains a comprehensive survey of the individual phases of resource management processes, focusing on resource modeling, resource discovery, resource estimation, and resource allocation approaches based on performance parameters or metrics, as well as on architecture types. This paper presents also the architecture of a generic resource management enabler. Furthermore, we present open issues concerning resource management, pointing out the directions of future research related to the Internet of Things.

**Keywords**—Internet of Things, resource allocation, resource discovery, resource management.

## 1. Introduction

The Internet of Things (IoT) connects physical world objects around us to the Internet. The heart of the IoT ecosystem comprises a huge set of smart devices with sensors and actuators, offering also a certain amount of computational and communication capabilities [1]. The operating principle of IoT devices or a system of smart objects is shown in Fig. 1. The system performs a number of tasks, such as sensing, communication, computation and actuation, which are required for collecting and processing data within the IoT environment. The sensors used may be of different types and may measure, for instance, temperature, humidity, pressure, etc. All such devices are used to gather data from their surrounding environments. In the next step, this information is uploaded to the server, either directly or via a gateway. Based on server feedback, IoT devices may send commands to actuators in order to properly adjust the specific parameters.

IoT finds numerous applications in the real-world environment, e.g. in smart homes, intelligent transportation, health

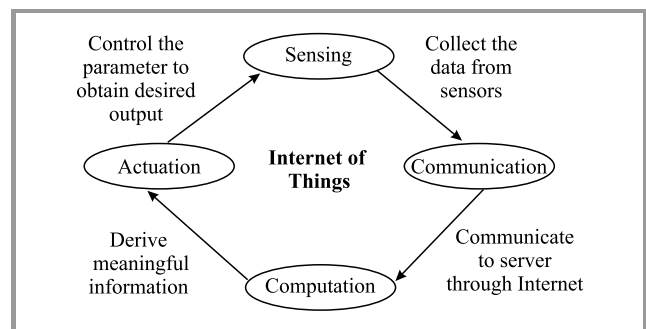


Fig. 1. Operating principle of IoT devices or smart objects.

monitoring, retail activities, smart cities, environment monitoring, power management, etc. [2], as shown in Fig. 2.

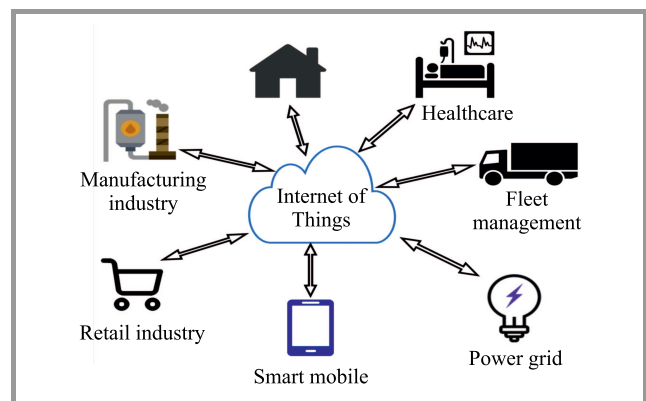


Fig. 2. Applications of IoT.

Due to the considerable benefits offered by IoT applications, industries, R&D labs and governments from across the world are investing huge amounts in IoT technologies. This has led to fragmentation and strong competition on the IoT market that suffers from incompatibility due to the existence of multiple standards. The International Telecommunication Union for Telecommunication (ITU-T) has recognized the different challenges faced by IoT networks in comparison to traditional networks. ITU-T has proposed

a new reference model for the IoT environment [3]. It consists of four layers: device layer, network layer, service support layer and application support layer. Management and security capabilities are taken into consideration as well. Since management and security capabilities apply to all four layers, they are represented vertically in the ITU's reference model.

The new paradigm of connected smart devices creates a number of research challenges, related mainly to interoperability across heterogeneous networks, machine-to-machine communication, self-aware and self-organizing networks, open framework for the IoT, large-scale deployment of infrastructure, dynamic, autonomous and adaptive resource management, ad-hoc deployable and configurable networks, distributed energy efficient data processing, mechanisms protecting against attacks, access control and accounting schemes for IoT, lightweight secure and high performance authentication, standardization of APIs, and context-aware adaptation of operation [4].

Traditional computer network architectures and reference models may not be suitable to tackle these challenges and issues. Many researchers have suggested an increased number of layers compared to TCP/IP, but none is standardized in the IoT environment (Fig. 3). The conventional three-layer architecture comprises application, network, and perception layers, whereas in the five-layer architecture, business, application, processing, transport and perception layers are distinguished.

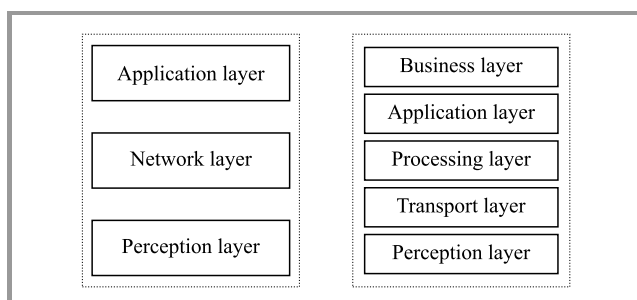


Fig. 3. Three- and five-layer architectures.

The application layer used in both architectures provides user specific services, such as health monitoring systems, fleet management, environment monitoring, and so on. The perception layer is responsible for managing sensors and actuators. It gathers such information as temperature, humidity, motion, etc., and forwards it to the network layer (in the 3-layer architecture) or to the transport layer (in the 5-layer architecture). It is responsible for networking the functionalities of sensing devices, gateways and the server. In the five-layer architecture, transport layer is responsible for transferring data using wireless technologies, e.g. cellular, Wi-Fi or Bluetooth, from the perception to the processing layer and vice versa. The processing layer is responsible for storing and analyzing data collected from the transport layer. It uses other technologies, such as cloud, fog or edge computing for processing such data. The busi-

ness layer manages the entire IoT ecosystem in order to provide services to the user.

In this paper, we focus on existing solutions and open issues related to managing resources in IoT ecosystems, i.e. on the basic functionality of IoT that needs to be supported in accordance with the ITU reference model. Research on IoT resource management is scarce, but the works available tend to focus on the stages of resource management processes, as well as on the classification of resource allocation methods based on energy, context, quality of service (QoS), service level agreement (SLA), load balancing, cost, efficiency, etc. [5]. WSN-based IoT networks, resource types and resource scheduler methods are taken into consideration as well, just as are resource allocation algorithms, integrated approaches to IoT and fog networks, as well as IoT, fog and edge devices.

But these works fail to discuss the overall nature of the process, e.g. what resources are required to perform the allocation. How it is modeled? Which types of infrastructures are used? How to take advantages of edge or fog computing in delay-sensitive applications? IoT networks integrate multiple heterogeneous networks, such as personal area network (PAN), mobile edge computing, and industrial IoT, thus creating new challenges. Information is lacking on how resources are identified, modeled and selected before the scheduling process. Taking into consideration the shortcomings of work that has already been performed, we have presented, in a systematic manner, the process of managing resources in IoT, the potential options available for modeling, and the existing algorithmic solutions. Finally, we also outline research-related challenges and open issues concerning IoT resource management.

In this work, we outline the importance of each phase of the resource management process and describe their interconnection with the reference model and SDOs. The main focus is placed on resource allocation algorithms and the link between this stage and the remaining phases of the process. The following are the most significant contributions of our work:

- systematic study of multi-tier architectures (and their stages) used for resource management,
- understandings different SDO and industry guidelines suggested for managing IoT resources,
- outline of existing resource modeling, resource discovery, resource estimation and monitoring techniques relied upon in IoT resource management,
- analysis of existing resource allocation approaches based on performance metrics and architectures,
- existing issues and future trends in IoT resource management.

The remaining part of this paper is organized as follows: Section 2 presents an overview of resource management

in IoT. The IoT resource management module used in Firmware is described in Section 3. Challenges and research-related issues concerning IoT resource management are outlined in Section 4. Finally, the discussion is concluded in Section 5.

## 2. Resource Management in IoT

This section describes the ITU-T reference model for IoT, ecosystems and the individual resource management stages. In an IoT ecosystem, the participating entities are heterogeneous in nature and rely on different processing capacities (8/16/32/64-bit), communication technologies (cellular/Bluetooth/Wi-Fi/ZigBee/RFID/LoRa) and storage capacities. The sensing devices may be deployed in a deep forest, may be a part of energy distribution systems used in cities, or may be installed in moving vehicles.

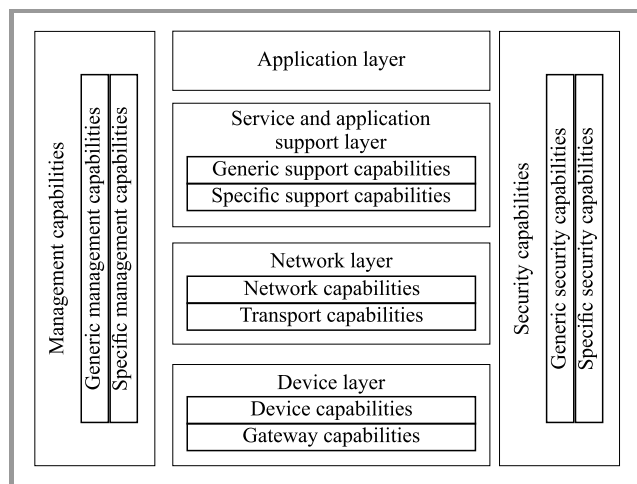


Fig. 4. ITU-T reference model for IoT.

Each IoT application or service – e.g. real time video streaming or simple event notifications – may have different requirements. Deploying, maintaining and monitoring a wide range of devices is a tedious task in a real world environment. In order to address heterogeneity and complexities of the network, ITU-T has proposed an IoT reference model shown in Fig. 4 [3]. The description of each layer is presented below:

- **Application layer** – comprises with end user IoT applications.
- **Service support and application support layer** – offers generic and specific support capabilities. Generic capabilities are used to offer common functionalities that may be used by any specific applications. Specific support capabilities provide services that are accessed by specific applications.
- **Network layer** – supports network transport capabilities. As far as network capabilities are concerned,

it supports device connectivity, mobility management, routing, authorization and authentication for diverse protocols. In terms of transport capabilities, it supports the transportation of application specific data, IoT services, control and management information.

- **Device layer** – this layer supports device and gateway capabilities. Device capabilities include direct interaction with the communication network, indirect interaction with the communication network, ad-hoc networking between devices, as well as sleep and wake-up cycle for energy conservation. Gateway capabilities include protocol conversion and multi interface support.
- **Security capabilities** – support for application specific and generic capabilities. General capabilities include authorization, authentication, user privacy protection, security audits in the application layer, secure communication, data and signaling at network layer, as well as authentication, authorization and access control over the device layer.
- **Management capabilities** – generic and specific capabilities are distinguished here. Specific capabilities deal with application specific requirements. Generic capabilities, in turn, perform functions for specific applications, including device activation, deactivation, remote access, device status, software updates, topology and traffic management. The management layer is responsible for fault detection, accounting, configuration, security and for performance of device and software components.

### 2.1. Resources in IoT Ecosystems

The various types of resources used in IoT environments are shown in Fig. 5. The resources may perform any tasks that are useful for the system, network or end user applications. In general, there are two main types of resource management approaches that are concerned with infrastructure and applications. Infrastructure-based resource management relates to computational, networking, storage and energy resource managements. From the system perspective, hardware (sensor, CPU, memory), firmware and software resources may be distinguished. From the network prospective, the resources may include the radio antenna, channel, bandwidth, routing path or nodes. Storage resource management is associated with various types of different components, such as memory, file system and so on. Similarly, energy resource management deals with energy, battery and so on. From the application or user perspective, resource management may affect any software components, such as application management module, resource information base, customer information base, resource identification or resource modeling.

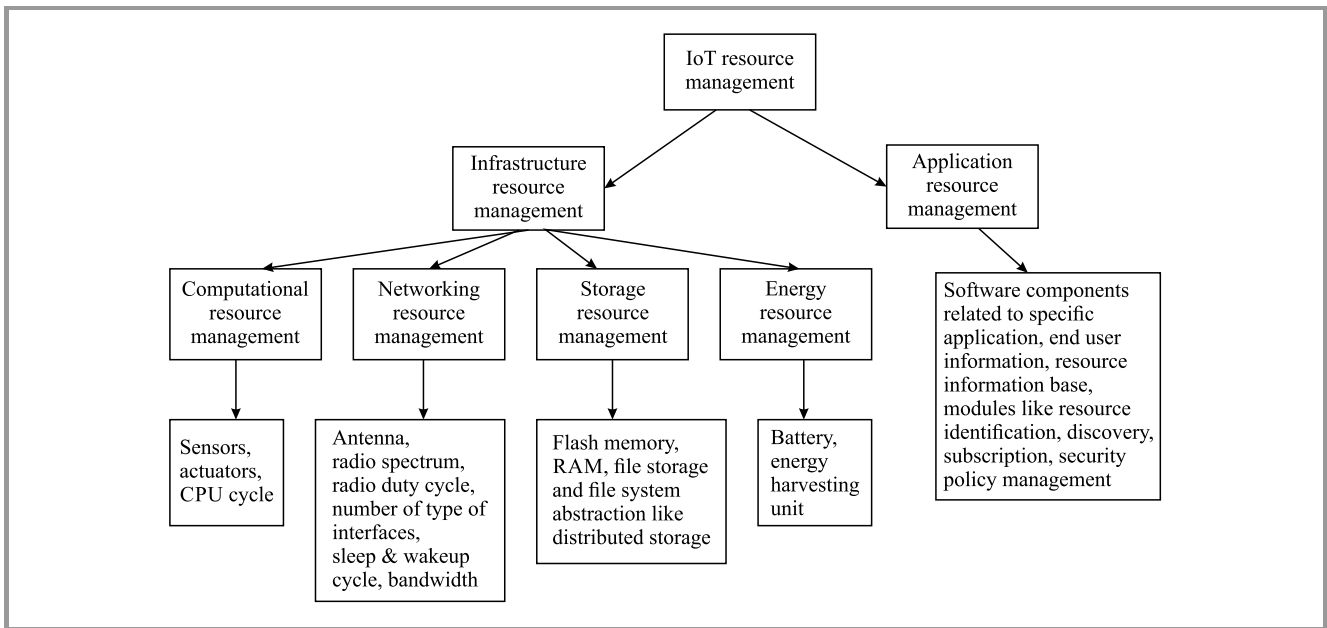


Fig. 5. Types of IoT resources.

2.2. Distributed Architecture of IoT Ecosystems

Data collected by sensor nodes has to be processed before being used for the provision of meaningful services to users or applications. Processing consumes multiple re-

sources that are geographically distributed. It involves heterogeneous hardware, software and communication technologies. Figure 6 shows the typical operating principle of a distributed IoT ecosystem.

Sensing devices may be IP or non-IP based. Non-IP based IoT devices are generally low powered devices using such protocols as Bluetooth, ZigBee and Wi-Fi. These devices forward the collected information to the nearest configured gateway. Data collected at each gateway is filtered and processed before being forwarded to the central server or cloud storage. IP-based IoT devices will forward the data either through the gateway, or they may upload it directly to the central server. Distributed architecture is associated with the following computing mechanisms that affect resources in the IoT environment:

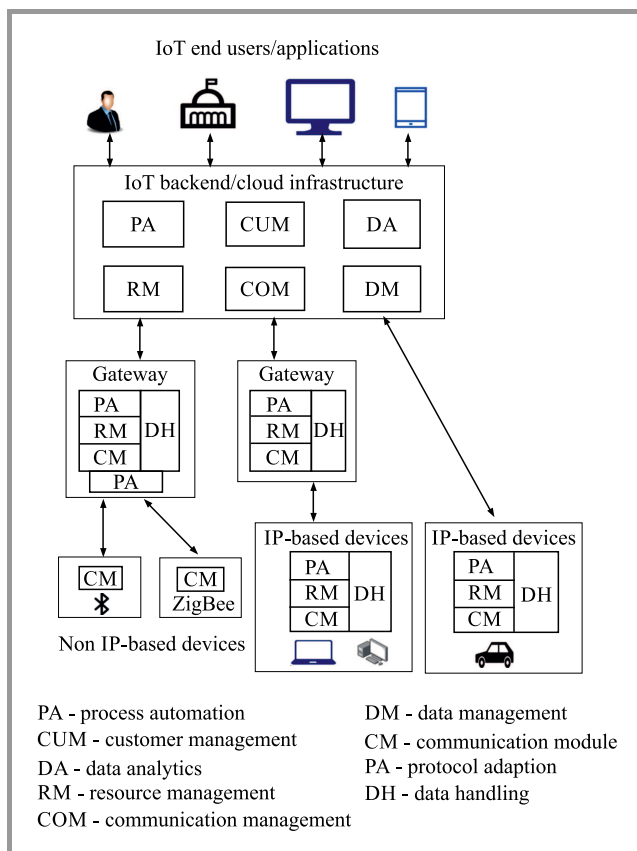


Fig. 6. Distributed architecture of IoT system.

- **Cloud computing resources in IoT ecosystems.** Clouds rely on vast computational and storage resources. IoT devices may be used in cloud computing infrastructures for storing and performing a complex analysis of IoT data. Cloud computing offers reduced storage, computational and operational cost. But the integration of IoT and cloud gives rise to many challenges [6]. Seamless Internet connectivity is required, mainly, and latency between cloud and IoT devices is also very high, which is unacceptable for real-time applications, such as health monitoring systems. Additionally, security and privacy will also pose major challenges, since data is stored remotely will also be a major challenge since data is stored remotely.
- **Edge computing resources IoT ecosystems.** Edge computing offloads the resource demand from end devices and cloud. Edge devices may include the

gateway, device controller or smartphone. Real-time applications that demand low latency and high response times benefit from edge computing. Edge computing contributes also to efficient context-aware computing and to mobility management.

- **Fog computing resources in IoT ecosystems.** The fog computing paradigm is similar to that of edge computing, bringing computation, storage and network services near the data source, instead of transferring them to cloud computing. But in fog computing, one or more devices in LAN help in offloading the workload of edge devices. This approach offers advantages that are similar to those of edge computing, but with a much higher processing and storage capacity.

Resource requirements of distributed IoT systems depend solely on the system's functional and non-functional requirements. Detailed specifications are provided by ITU-T [7]. The different requirements and their classification are presented below:

- business requirement specification – a high-level document that specifies the organization's motivation to development of a new system,
- stakeholder requirement specification – includes the perspective and requirements of different stakeholders, such as users/operators/maintenance personnel, related to using the system to generate business contributions,
- system requirements specification – specifies the technical requirements of the selected system, including those of functional and non-functional character:
  - **functional requirement specification** – specifies the behavior of the system in terms of service/function/operation. The important functional requirements specified by ITU for IoT networks include application support requirements, service requirements, communication requirements, device requirements, data management requirements, security and privacy protection requirements,
  - **non-functional requirement specification** – specifies supplementary/quality-related characteristics of the system. The important non-functional requirements specified by ITU-T for IoT networks include interoperability, scalability, reliability, high availability, adaptability and manageability. Non-functional resource requirements are difficult to calculate, as they are of the subjective and relative nature [8]. An attempt to improve one parameter may exert a direct impact on other parameters of the same system.

In the IoT environment, management of resources is a crucial task in terms of fault tolerance and security. The resource management functionality is distributed across the different layers of the reference model. Any hardware, software and communication modules can be modeled as resources.

### 2.3. Resource Management in IoT

The individual stages of the resource management process are shown in Fig. 7. Resource modeling is an important stage in resource management. It is the process of abstracting and building metadata for IoT resources. Effective resource modeling is the key to efficient resource discovery and allocation mechanisms in IoT networks. Resources are modeled based on semantic, virtualization and attribute modeling methods. Each of these modeling methods is described below:

- **Semantic modeling.** This model helps build loosely coupled, interoperable and service-oriented IoT architectures. Device capabilities or services rely on standard or machine-readable formats, which enhances machine-to-machine (M2M) communication, derivation of new knowledge and interoperability [9]. Different SDOs are working to address issues related to standardization. Sensor web enablement by Open Geospatial Consortium (OGC) addresses mainly sensors, sensor data models and sensor web services [10], [11]. It also includes a specification related to observation and measurement (O&M), sensor modeling language (SML), transducer model language (TML), sensor observation services (SOS), etc. W3C defines also the formal specification for data representation by means of a resource description framework (RDF) and ontology representation language, such as ontology web language (OWL).
- **Virtualization based modeling.** In this model, sensor hardware, software or network resources are represented virtually in a cloud or a gateway, for ease of management and for optimized resource utilization [12]. Different types of virtualizations relied upon in IoT are listed below:
  - **Software defined network (SDN).** SDN abstracts the physical network, which helps in controlling and configuring the network behavior centrally. SDN has two planes – one is the forward plane for packet transmission, and the other is the control plane for network configuration and management. Decoupling these planes aids in element of heterogeneous devices;
  - **Network function virtualization (NFV).** NFV eliminates the need for dedicated hardware and uses a commodity server for network functions. Different network functions, such as routing,

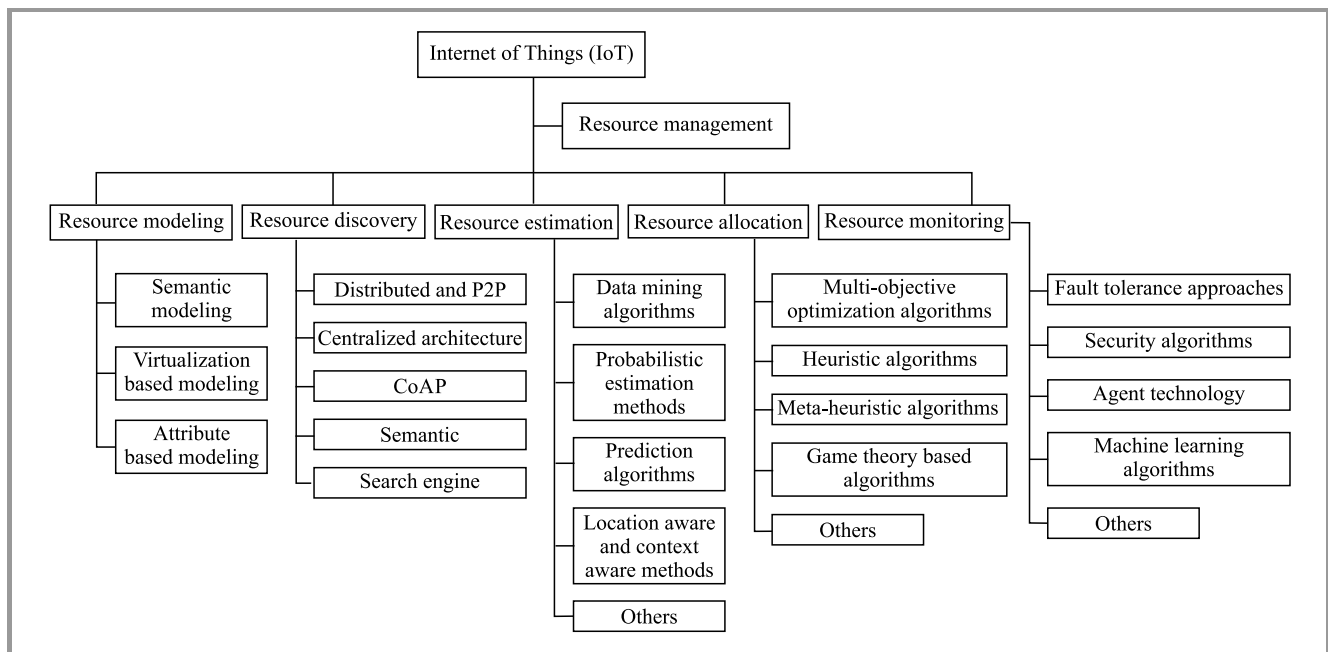


Fig. 7. Stages of the resource management process.

firewall, load balancing and intrusion prevention may be performed by the commodity server due to NFV implementation;

- **Containers/dockers.** This approach is an OS level virtualization, where applications are packed with their own dependencies, libraries and configuration files. Since multiple containers are executed directly on the host OS, the solution is more lightweight than virtual machines. In IoT ecosystems, many cloud service providers support development and management of containers [13]. Containers are generally deployed in edge devices, helping in real time data analysis (e.g. packet inspection and intelligent alert systems). They also act as load balancers by reducing the load of cloud devices;
- **Sensor virtualization.** In this approach, an abstract model of IoT devices is hosted on the central server. End user applications submit requests to the central server, with an abstract IoT device representation, rather than to physical IoT devices. Sensor virtualization enhances energy efficiency and lowers maintenance cost [14].
- **Attribute based modeling.** In attribute based resource modeling, resources are designed based on functionality and attributes of devices, such as their location, energy, sensing parameters, etc.

Results of some research projects related to resource modeling have been published recently. In [15], the authors suggest a set of tasks to be performed by a set nodes in polynomial time. While modeling the metadata, the author

has considered various properties and constraints of nodes. Such constraints as location, sensing capacity, power, software and hardware are modeled in the form of a binary programming model and are then applied using the heuristic method. In paper [16], the author modeled the problem of assigning different IoT services to different physical interfaces. In this approach, interfaces are modeled as resources. Since the interfaces are limited in any system, multiple services compete for those interfaces that are available. The author proposes two algorithms: single round assignment – when sufficient resources are available, and multiple rounds – when resources cannot be shared in the single round. Service splitting is performed to handle the workload. Splitting and distribution cost is also considered. Another approach is based on application demand and resources are identified based on the deployed WSNs [17]. Then, the linear programming model is applied to map the resources and application requirements. It considers different attributes, such as functional capability of the node, granularity of operation, power consumption, location, duty cycle and fidelity region. In [18], the authors discuss hardware, software and communication resources that are integrated to provide services (i.e. everything as a service) to IoT user. They also place a greater emphasis on information rather than on infrastructure.

The work presented in [19] has led to designing an IoT service discovery and ranking method that uses semantic modeling. Ontology linking is used to create semantic reasoning for better IoT services. Semantic based resource modeling approaches are effective in handling heterogeneous, distributed, and ultra large-scale resources in IoT. Paper [20] presents the OpenIoT platform that uses W3C SSN ontology. It also explains the process of implementing IoT applications with cloud integration. The W3C SSN on-

Table 1  
Comparison of different resource modeling techniques

References	Attribute-based approach	Semantic-based approach	Virtualization-based approach	Method used	Proposed approach
[15]	•	○	○	Binary programming model is developed to represent attributes of IoT device and its constraints.	Heuristic method to improve execution time and load balance.
[16]	•	○	○	Mathematical model considers services and interfaces as resources.	Mapping of interfaces to services improves the utilization, service, and activation cost.
[17]	•	○	○	Deployed WSNs resources are modeled as a resource map for satisfaction of IoT requirements.	The framework optimizes the resource mapping and allocation time.
[18]	○	•	○	Hardware, software and network are modeled as a service.	A new service model where hardware, software and communication modules are integrated as a single service for the user.
[19]	○	•	○	Ontology linking is performed for service discovery and ranking.	Scalable solution based on semantic reasoning.
[20]	○	•	•	SSN ontology is used for middleware, supporting virtual sensors in cloud.	The model is based on the W3C SSN ontology which provides better interoperability.
[21]	○	•	○	A new, SSN ontology-based design pattern is proposed.	Lightweight semantic modeling method, extendable due to linked data.
[22]	•	•	○	NFV is applied at intermediate nodes.	Two intermediate processing nodes for better execution time and response time.

tology is described from the perspective of sensors, sensing capabilities, stimuli and observations [21]. The work proposes also the stimulus-sensor-observation ontology design pattern, which relies on light-weight semantics preferred by several linked data applications.

The importance of semantic interoperability and its challenges are presented in [22]. The author discusses about challenges related to data modeling, information exchange, semantic annotation and semantic discovery of data sources, semantic reasoning and interpretation, as well as knowledge representations. The work outlines also a few important requirements for IoT semantic platforms. In paper [23], the author quantitatively evaluated four different IoT middleware platforms using a small-scale IoT scenario. In this work, a comparison of functional components, such as service registration, discovery and composition, was performed. The work highlights the need for an autonomous system with automatic service registration, discovery and composition to handle heterogeneous devices in IoT environments.

Virtualization of IoT devices for such applications as smart cities is discussed in [24]. Two intermediate nodes are added to the system, i.e. an application node and a function node. Gateway nodes store raw data generated by the sensors, whereas function nodes process the raw data. A service application connects directly with the function nodes.

This work uses NFV to service a node, rendering flexible and scalable micro-services. The proposed prototype is integrated – via cloud, edge and IoT devices – with a camera. Results of the experiment prove reduced processing time and improved resource sharing.

Different resource modeling approaches are summarized and compared in Table 1. One may see that efficient resource modeling approaches are very important to achieve portability, scalability and ease of maintenance. Semantic based modeling facilitates M2M communication and process automation. But due to multiple SDOs and their formats, interoperability is still a challenging task. Also, while building resource models, the level of abstraction should be selected legitimately. Otherwise, metadata management will become a tedious task. Resource modeling based on the attributes of resources may be used to build mathematical models for simulating and testing real world scenarios. As far as virtual resource modeling is concerned, it helps ease maintenance and offers better utilization of resources. Virtualization based SDN and NFV aid in handling heterogeneity and scalability.

#### 2.4. Resource Discovery in IoT

The resource discovery module is responsible for identifying the resources required to satisfy application requests,

i.e. for device registration, configuration and ranking. Due to the large number of different IoT devices, the resource discovery module is expected to offer automatic discovery of resources, their properties and capabilities. Some of the existing technologies used here include P2P and distributed resource discovery, centralized architecture, CoAP-based, semantic based, and search engine based resource discovery. These are described below:

- **Distributed and P2P Discovery Services.** In this method, distributed devices are used to adopt a distributed hash table and are interlinked by a P2P overlay protocol to enable local and global service discovery. The method supports also multi attribute indexing, range queries and P2P routing. Therefore, the distributed architecture model effectively handles scalability and various issues related to IoT;
- **Centralized Architecture for Resource Discovery.** In this method, a central infrastructure is built for resource discovery. All devices should register to the centralized infrastructure before providing services. Client devices may discover and take advantage of a given service by accessing the central server;
- **CoAP-based Resource Discovery.** Here, the service discovery mechanism is adopted. CoAP servers expose a set of RESTful web services to clients. The interested CoAP clients request the server to provide service. Based on the request, a client may receive a list of resources and their attributes or metadata. But this method has certain drawbacks related to resource discovery lookup by remote devices. It also suffers from scalability and security issues;
- **Semantic-based Resource Discovery.** Resource discovery is based on semantic interpretation of resources. Standard web technologies are used for semantic representations of devices and their capabilities,

fostering interoperability, such as URI, HTTP, SensorML, JSON (JavaScript Object Notation) and efficient XML Interchange (EXI);

- **Search Engine-based Resource Discovery.** This model is similar to the Google search engine, but is not identical. The Google search engine is based on web documents, while an IoT search engine is based on resources, attributes and their properties. There are different types of IoT search engines, such as keyword, location, real-time, and hybrid search engines.

Paper [25] presents a search engine based resource discovery framework which is made up of three layers, namely proxy, discovery and service enablement layer. The proxy layer discovers the objects regardless of their technologies. It includes drivers and the protocol required to communicate with the devices. The discovery layer is responsible for storing metadata in the CoRE link format and for searching and ranking the resources based on user input. The service enablement layer provides the service to the user by RESTful web services. Interoperability between different description models of the sensor is a challenging task. Paper [26] describes a new framework based on keyword or geospatial searching approach. This framework supports API, SensorML and W3C JSON-LD description models. It will automatically detect the modeling language and parse its data. Then, this parsed data is converted into a standard format and stored in the database, along with location. In [27], a centralized resource discovery framework is proposed, using web standards. In the proposed framework, resource categorization and indexing are done based on multiple domains. SDN based resource discovery in IoT networks is discussed in [28]. It suggests that traditional network architectures are not suitable for IoT resource discovery and routing path definition. To obtain a congestion free network, the author proposes an SDN based IoT net-

Table 2  
Comparison of different resource discovery approaches in IoT

References	Method/approach	Centralized	Distributed	Framework/prototype/model proposed
[25]	Search engine based	•	○	New framework is proposed with proxy, discovery and service enablement modules.
[26]	Semantic-based indexing	•	○	The proposed framework parses metadata of multiple standards and creates indexes automatically.
[27]	Semantic-based	•	○	The proposed framework stores context-aware and geolocation information, along with resource information.
[28]	SDN-based	•	○	iFogSim simulation tool is used for efficient resource discovery based on SDN.
[29]	Agent-based	○	•	Multiple agents are installed in IoT devices, gateway and server.
[30]	Mathematical model	○	•	Effects of multiple context aware parameters are analyzed using a mathematical model.



work. Context information from network entities, computational devices and IoT devices is analyzed by means of a centralized SDN approach. The proposed model is simulated using the iFogSim tool which shows that SDN based IoT networks outperform traditional networks.

An autonomous multi-agent resource discovery mechanism for IoT is discussed in [29]. The author proposes a model with multiple agents running on the server, gateway and each IoT device. Initially, the server agent forwards the command to the gateway agent that forwards the command to agents of the particular devices. The proposed work assumed that agents are already preinstalled in each IoT device. This allows to avoid any commands that may be potentially missing during sensor sleep schedule, and IoT devices register their sleep pattern and type of service with the gateway agent.

The work presented in [30] analyses the different multi-objective decision making methods. The model is based on the centralized resource discovery model which uses the number of sensors and context information to ensure efficient results. The author claims that results obtained with the use of the simple additive weight method (SAW) offer better quality compared to the technique relying on the prioritization order, with that quality being similar to an ideal solution. In [31], the authors compare three commonly used discovery protocols, namely CoAP, MQTT and UPnP, based on memory footprint, CPU footprint, latency, downstream and upstream traffic. In this work, results of an experiment performed show that CoAP outperforms MQTT and UPnP, but at a higher memory overhead.

Table 2 summarizes different resource discovery approaches presented in recent publications.

### 2.5. Resource Estimation

In most IoT applications, user requests and data generation are spontaneous processes that pose a great challenge in the prediction of IoT resources. But for optimal system performance and efficient utilization, a certain level of estimation is not accurate enough. Generally, estimation of resources is performed based on historical records stored in the cloud. Fog may also aid in the estimation of resources, as they are deployed near to the sensing devices and they also have better knowledge of the location and higher context awareness.

IoT resource estimation depends on device type, device mobility, energy status, type of data generated or processed, communication method, security measures adopted and customer behavior. The extra pre-allocation of resources may lead to their underutilization. Improper allocation of resources will also directly affect system performance. Therefore, the resource management system should accurately predict the resources required for optimal performance of the system. In this regard, data analytical tools can be used for the prediction of data based on historical records stored in the cloud. The context information derived from edge and fog devices can also be used for accurate estimation of nearby resources. Data

mining algorithms, probabilistic methods, predictive algorithms, location- and context-aware resource approaches, etc. are used for this purpose.

Some research papers are devoted to IoT resource estimation. In [32], a mathematical model for determining customers' reliability and loyalty in order to ensure better utilization of resources is proposed. In this work, relinquish probability is used to prioritize loyal customers and for giving a fair chance to new ones. [33] proposes a resource estimation and allocation approach that is based on customer and device type. In the model, customers are categorized as new, absolutely new and existing. Paper [34] presents a resource allocation method based on customer type and loyalty. It uses crawled real trace and Amazon EC2 pricing. Results show that loyal customers are provided with better service and resources are allocated cautiously to other customers.

### 2.6. Resource Monitoring

IoT networks link a vast number of heterogeneous devices that support a wide range of applications. Real time applications, e.g. those used for patient health monitoring, flood alerting, fire alarms and for reporting loss of or delays in transferring information, may be a cause of a serious failure. Monitoring tools used in IoT should detect device failures, degraded performance, network delay, security attacks, etc. Algorithms are also needed for resource monitoring, (fault tolerance algorithms, security algorithms, agent-based algorithms, machine learning and so on).

In [35], the challenges related to IoT resource monitoring, e.g. identification of sensor malfunctions, inappropriate calibration, delays, packet loss, network failures, device energy status, security attacks and performance issues related to CPU and memory are discussed. [36] presents a new IoT monitoring tool for the agent-manager model. It is a lightweight solution compared with the SNMP based monitoring system. The agent devices follow commands forwarded by the manager device, i.e. a gateway.

Paper [37] outlines the IoT ecosystem challenges that are related to effective monitoring of resources. It also compares two standard network monitoring tools, namely big brother (BB) and Zenoss.

### 2.7. Resource Allocation

In a heterogeneous IoT system, achievement of a satisfactory level of service (SLA) related to end-user applications is a major challenge. For example, patient monitoring systems and city surveillance systems require completely different resource management strategies. When an application requests a service from IoT devices, multiple steps are performed before the resources are allocated. In short, client authentication requires specific resources, so the pool of available resources, access permission and utilization constraints need to be verified in the middleware framework. To solve complex resource allocation problems,

mathematical models are required that operate in the real world environment. In such models, the specific scenario is mapped to a standard optimization algorithm. The important categories of resource allocation algorithms include the following:

- **Multi-objective optimization algorithms.** These algorithms identify the best option from a set of feasible solutions. IoT resource allocation problems are often modeled as multi-objective optimization problems, since they involve a conflict of multiple criteria objectives. Most resource allocation problems try to apply the Pareto rule, where a change in any objective degrades at least one of the remaining objectives. Since these algorithms find exact solutions, they are computationally complex and consume more computational power;
- **Heuristic algorithms.** These algorithms are faster in finding an approximate solution instead of an accurate solution that requires a higher computation cost. They are also used to obtain near optimal solutions when other optimization algorithms fail to generate exact data. All these benefits are achieved by trading optimality, accuracy, precision and completeness of the solution. These algorithms lack in flexibility and are sensitive to key decision parameters. Even minor changes will drastically affect the end solution. Examples of such algorithms include, greedy and tabu search approaches;
- **Meta-heuristics algorithms.** These are higher level procedures relied upon to find a sufficiently good solution based on incomplete or imperfect information. These algorithms do not provide a globally optimal solution. Their examples include ant colony optimization, particle swarm optimization or genetic algorithms;
- **Game theory-based algorithms.** Game theory models rely on rational behaviors of two or more players and on their interdependence. Cooperative and non-cooperative games models are usually applied to solve resource allocation issues in IoT ecosystems by achieving the Nash equilibrium. The game theory model may not be suitable for a number of scenarios. For example, the game theory expects each player to be homogenous and to have similar capabilities, while IoT networks consist of highly heterogeneous devices. Also, it expects multiple interactions between the participating devices. Fulfillment of this condition is not feasible due to the large number of devices and the related energy constraints.

Among many papers focusing on resource allocation, it is worth mentioning [38] which discusses network lifetime and throughput for uplink communication in battery powered IoT networks. At first, the available channels are grouped and allocated to IoT users. Then, transmission

power allocation is optimized for each user group based on the Markov decision process (MDP). The efficient channel allocation algorithm (ECAA) proposed outperforms the random channel assignment approach by 90% and offers better time efficiency.

The problem of assigning a gateway to IoT devices is discussed in [39]. Each sensing device has to forward data either to the cloud or the server through the one or more gateways. Proper mapping of IoT devices and gateways allows to achieve good QoS. In the proposed work, the author modeled the gateway allocation problem as an NP-complete bipartite matching problem, where IoT devices have different data transmission requirements. Later, the work proposes a repetitive Ford-Fulkerson algorithm (RFF) for equal workload scenarios, and an approximation algorithm called greedy gateway algorithm (GGA) for unequal workload situations, minimizing the maximum traffic at the gateway.

Paper [40] presents a resource allocation problem experienced in the downlink connection of a fog-based IoT network. QoS parameters, such as delay, throughput and bit error rate are considered for ultra-reliable low latency communications (URLLC) and for enhanced mobile broadband (eMBB) service type applications. URLLC requires a high tolerable BER, whereas eMBB requires a high bandwidth which may comprise delay and BER. Centralized resource allocation will be NP-hard considering the heterogeneous nature and the number of devices involved. The integrated analytic hierarchy process (AHP) and the two-sides matching game approach ensure higher utility gains. AHP reduces complex QoS into traceable hierarchical subproblems, which helps in prioritizing QoS parameters, while the two-side matching game approach is initiated between fog network infrastructure and IoT devices. The service-to-interface assignment (SIA) problem that is extended onto multiple IoT devices and modeled as SIA-MID (service-to-interface assignment for multiple IoT devices) is presented in [41]. The classic SIA problem is designed for a single IoT resource, whereas SIA-MID is designed for more than one IoT device.

Bandwidth allocation between access points (APs) and the base station (BS) and modeled as a hierarchical two-stage game model is presented in [42]. At the first stage, bandwidth allocated to BS is distributed among APs using the asymptotic Shapley value method. Each AP acts as a game player forming an inter-AP game. In the second stage, bandwidth allocated to each AP is distributed across different classes of multimedia services using the relative utilitarian value approach. Different classes of traffic services will act as game players here. At the end of each unit's time period, the relative utilitarian value is recalculated and bandwidth is redistributed across APs. When compared with the game-theoretic hierarchical resource allocation (GHRA) scheme and the auction-based hierarchical resource allocation (AHRA) model, the algorithm shows better throughput, service rate and minimal resource utilization.

[43] proposes a new resource allocation mechanism called search economics-based IoT resource allocation (SEIRA) and using meta-heuristic and data clustering algorithms. In the data clustering algorithm, k-means is used to generate improvised initial values that are used in search economics (SE). SEIRA uses hybrid encoding, modified transit and a local search operator for better results. SEIRA and SEIRA-KD (SEIRA with k-means and dynamic local search) yield better results and are faster compared with GA and SA methods. An optimal resource allocation strategy for the three-tier IoT architecture is proposed in [44]. The three-tier architecture includes data service operator (DSO), data service subscribe (DSS) and fog nodes (FN), where each entity acts autonomously and observes the behaviors of each other for better decision making. The proposed algorithm has four stages, including resource purchasing problem for DSSs, with DSOs modeled as the Stackelberg game, pricing problem for the DSOs, DSO and FN pairing problem as many-to-many matching games, and under same DSOs pairing problem between FN and DSS as many to many matching problems. Simulation results show that DSOs, FNs, and DSSs achieve higher utility rates autonomously.

A distributed consensus-based algorithm in which virtual objects (VOs) negotiate with other elements in order to optimize resource allocation by distributing the workload is presented in [45]. It extends the VOs model to incorporate quality of information (QoI) for decision making. The result shows an improved node lifetime and a faster convergence time for task distribution. The fog radio access network (F-RAN) resource allocation problem is discussed in [46]. Fog RAN reduces latency – an important parameter in IoT applications. The proposed algorithm is modeled as MDP and is further reinforcement by learning to obtain the optimum solution. Simulations of the proposed algorithm outperforms the fixed threshold algorithm.

Paper [47] proposes a middleware-based approach used to distribute the task among different IoT devices, so that resources are adequately shared. The proposed middleware has two layers, i.e. semantic layer for interoperability of nodes, as well as resource allocation and management layer (RAML) for task group management. The distributed protocol is proposed based on the consensus algorithm which converges the optimal allocation of resources offering, in most cases, an error rate of less than 5%. The proposed model is tested in Matlab, using the Telit development board. Data transmission between gateways is presented in [48]. Composite services are executed at the gateway by using one or more IoT resources that are allocated to at least one gateway. To reduce the volume of data exchanged between gateways while executing composite services, network topology is transformed into the degree-constrained minimum spanning tree (DCMST). A modified version of DCMST is used to identify the optimal solution by applying the genetic algorithm. Simulations of the proposed algorithm show a 97% success rate, on average, while finding a near optimal solution. Allocation of heterogeneous

IoT resources is modeled as the Stackelberg game model in [49]. The network operator is modeled as the leader and the mobile terminal as a follower. The network operator sells the resources to the mobile terminal based on dynamic pricing. A distributed iterative algorithm is proposed which allows the entire network to reach the Nash equilibrium as a part of the Stackelberg game model. A simulation result provides an optimal pricing strategy.

The work presented in [50] proposes an adaptive resource allocation algorithm (AMRS) for IoT devices with constrained bandwidth. When performing sensing and actuating operations, network bandwidth is dynamically assigned based on the signals' frequency domain characteristics. Compared with the static algorithm, AMRS reduces accumulated and maximum errors by 60%, while path tracking the UGV. A novel resource management framework is proposed based on the device cloud approach [51]. In the proposed framework, IoT resources are considered as part of a cloud resource pool. The proposed work considers IoT end devices as data end points (DEP), the gateway as a data integration point (DEP), and the communication infrastructure as a machine communication network (MCN). The core components of the proposed framework include device directory which provides a data model for the devices and their life cycle. The user directory module is designed for identifying and controlling access to management services. The management module is developed for performing the resource allocation process. The proposed framework is implemented at the gateway or an aggregating node. Article [52] discusses the challenges that need to be tackled in the device cloud approach, such as runtime device integration, interoperability, data models, nomenclature, and the service execution environment.

Paper [53] offloads the task from the local wireless device to a fog network. The different parameters to be managed at the device and network level are modeled using mixed integer linear programming (MINLP). The hybrid genetic simulated annealing algorithm is applied to solve the proposed issue, resulting in lower power consumption by local devices, faster convergence rates and minimum latency. An energy efficient resource optimization algorithm for cognitive IoT (CIoT) is proposed in [54]. Since CIoT consumes more energy for sensing operations in order to achieve better spectrum efficiency, the proposed algorithm harvests the primary user's RF energy to increase network lifetime. The number of nodes, sensing time and transmission power are modeled as a non-convex optimization problem. The proposed work suggests a node alternative algorithm which uses Lagrange optimization for power adjustments and Dinkelbach's optimization for node energy and network lifetime corrections.

Paper [55] presents an application-aware workload distribution approach focusing on edge devices. Resources of edge devices are dynamically adjusted to reduce response time. A heuristic algorithm is applied to distribute user requests to different edge devices. Research presented in [56] proposes a framework for multiband spectrum sensing,

Table 3  
Comparison of different resource allocation approaches

References	Mathematical model (MM)/ prototype framework (PFW)	Resource allocation infrastructure	Scalability	Heterogeneity	Load balancing	Real time	QoS	Fault tolerance	Mobility	Performance
[38]	MM – Markov decision process	<ul style="list-style-type: none"> <li>gateway with LoRa</li> <li>IoT device</li> </ul>	○	○	○	●	○	○	●	<ul style="list-style-type: none"> <li>network lifespan</li> <li>throughput</li> </ul>
[39]	MM – greedy method (heuristic method)	<ul style="list-style-type: none"> <li>IoT device</li> <li>gateway</li> </ul>	●	○	●	○	○	○	○	<ul style="list-style-type: none"> <li>traffic minimization</li> </ul>
[40]	MM – game theory	<ul style="list-style-type: none"> <li>fog devices</li> <li>IoT device</li> <li>gateway</li> </ul>	●	●	○	○	●	○	○	<ul style="list-style-type: none"> <li>throughput</li> <li>utilization of fog devices</li> </ul>
[41]	MM – metaheuristic method	<ul style="list-style-type: none"> <li>IoT device</li> <li>gateway</li> </ul>	●	●	●	○	○	○	○	<ul style="list-style-type: none"> <li>faster convergence</li> </ul>
[42]	MM – game theory	<ul style="list-style-type: none"> <li>access point</li> <li>IoT device</li> <li>gateway</li> </ul>	●	○	●	●	●	○	●	<ul style="list-style-type: none"> <li>throughput</li> <li>service rate</li> <li>better resources utilization</li> </ul>
[43]	MM – metaheuristic method	<ul style="list-style-type: none"> <li>IoT device</li> <li>gateway</li> </ul>	●	○	●	○	○	○	○	<ul style="list-style-type: none"> <li>faster solution convergence rate</li> </ul>
[44]	MM – game theory	<ul style="list-style-type: none"> <li>IoT device</li> <li>fog network</li> <li>data service provider network</li> </ul>	●	○	●	○	○	●	○	<ul style="list-style-type: none"> <li>higher utilization at fog and IoT devices</li> </ul>
[45]	Framework based on distributed consensus-based algorithm	<ul style="list-style-type: none"> <li>cloud</li> <li>IoT devices</li> </ul>	●	○	●	○	○	○	○	<ul style="list-style-type: none"> <li>node lifetime</li> <li>faster convergence</li> </ul>
[46]	MM – Markov decision process	<ul style="list-style-type: none"> <li>fog node</li> </ul>	●	●	○	●	○	○	○	<ul style="list-style-type: none"> <li>latency</li> <li>utilization</li> </ul>
[47]	Middleware framework prototype	<ul style="list-style-type: none"> <li>cloud</li> <li>IoT device</li> </ul>	●	○	●	○	○	○	○	<ul style="list-style-type: none"> <li>faster convergence</li> <li>less error rate</li> </ul>
[48]	MM – metaheuristic method	<ul style="list-style-type: none"> <li>IoT</li> <li>gateway</li> </ul>	○	●	●	○	○	○	○	<ul style="list-style-type: none"> <li>faster convergence</li> <li>minimize traffic</li> </ul>
[49]	MM – game theory	<ul style="list-style-type: none"> <li>network operator infrastructure</li> <li>IoT device</li> </ul>	●	●	○	○	○	○	○	<ul style="list-style-type: none"> <li>utilization</li> </ul>
[50]	MM – multi objective optimization	<ul style="list-style-type: none"> <li>IoT device</li> </ul>	○	○	○	●	●	○	●	<ul style="list-style-type: none"> <li>error rate reduction</li> </ul>
[53]	MM – metaheuristic method	<ul style="list-style-type: none"> <li>fog devices</li> <li>gateway</li> <li>IoT device</li> </ul>	○	●	●	○	○	○	○	<ul style="list-style-type: none"> <li>node energy</li> <li>latency</li> </ul>
[54]	MM – heuristic method	<ul style="list-style-type: none"> <li>gateway</li> <li>IoT device</li> </ul>	○	○	●	○	●	○	○	<ul style="list-style-type: none"> <li>network lifetime</li> <li>node energy</li> </ul>
[55]	MM – heuristic method	<ul style="list-style-type: none"> <li>fog devices</li> <li>gateway</li> <li>IoT device</li> </ul>	●	●	●	○	○	○	○	<ul style="list-style-type: none"> <li>latency</li> </ul>
[56]	MM – heuristic method and game theory	<ul style="list-style-type: none"> <li>gateway</li> <li>IoT device</li> </ul>	○	●	○	○	●	○	○	<ul style="list-style-type: none"> <li>faster convergence</li> </ul>
[57]	MM – game theory	<ul style="list-style-type: none"> <li>gateway</li> <li>IoT device</li> </ul>	●	●	○	○	●	○	○	<ul style="list-style-type: none"> <li>node energy</li> <li>packet to data ratio</li> </ul>
[58]	MM – metaheuristic method	<ul style="list-style-type: none"> <li>gateway</li> <li>IoT device</li> </ul>	●	○	●	○	○	○	○	<ul style="list-style-type: none"> <li>service rate</li> <li>node energy</li> </ul>
[59]	Block chain based framework	<ul style="list-style-type: none"> <li>gateway</li> <li>IoT device</li> </ul>	●	●	○	○	○	○	○	<ul style="list-style-type: none"> <li>computation cost</li> </ul>
[60]	MM – heuristic method	<ul style="list-style-type: none"> <li>fog devices</li> <li>gateway</li> <li>IoT device</li> </ul>	●	●	●	○	○	○	○	<ul style="list-style-type: none"> <li>service rate</li> <li>utilization</li> </ul>
[61]	MM – heuristic method and game theory	<ul style="list-style-type: none"> <li>gateway</li> <li>IoT device</li> </ul>	●	○	○	●	●	○	○	<ul style="list-style-type: none"> <li>data rate</li> <li>node energy</li> </ul>
[62]	MM – heuristic method	<ul style="list-style-type: none"> <li>gateway</li> <li>IoT device</li> </ul>	●	○	○	○	●	○	○	<ul style="list-style-type: none"> <li>data rate</li> <li>node energy</li> </ul>
[63]	MM – metaheuristic method	<ul style="list-style-type: none"> <li>fog devices</li> <li>gateway</li> <li>IoT device</li> </ul>	●	○	●	○	○	○	○	<ul style="list-style-type: none"> <li>energy</li> <li>utilization</li> </ul>

using the branch and bound algorithm. It provides a cross-layer configuration scheme for channel allocation based on the potential game. The proposed algorithm helps achieve dynamic resource allocation and also improves QoS parameters, such as energy consumption, throughput, delay, reliability and cost. In [57], the author provides energy efficient and cost-effective solutions for threat models using Pareto and Stackelberg leadership games. In the suggested model, the packet-to-data ratio (PDR), energy and the cost of installation are improved. The proposed model is tested using the TOSSIM simulator.

Article [58] proposes a task offloading and resource allocation model for mobile edge computing in an unmanned aerial vehicle (UAV). The proposed application involves resource intensive tasks that need to be performed with minimum delay, which is an additional constraint. In the proposed work, UAV position, uplink, downlink and computation resources are modeled as an optimization problem. Simulation results show a reduced service delay and lower energy consumption by IoT devices. In [59], a Blockchain based resource allocation approach is proposed, relying on edge computing. Edge computing devices act as a service provider for mobile devices which are willing to run their miners. In the model presented, a new hash-based power function is defined and an auction is conducted to reduce computational cost and to increase social welfare. [60] presents a method for balancing the load of fog devices. To reduce the load, services are migrated dynamically. The proposed model is tested on CloudSim, showing improved resource utilization, service rate and a reduced number of service nodes required. Paper [61] improves the data rate and energy utilization by using the heuristic greedy method and the cooperative game theory in the IoT network. The greedy method is used to find the optimal local solution, whereas interference between radio signals generated by IoT devices is reduced by means of the game theory.

For adaptive carrier spectrum allocation and joint spreading time code generation, the joint step extreme recursion (JSER) algorithm is proposed in [62]. This anti-jamming self-adaptive algorithm reduces power consumption and improves the signal to interference plus noise ratio (SINR). The computational and cache resource management problem is modeled as a joint optimization problem in [63]. The SDN-based information-centric model takes the allocation decision based on the Q-deep learning artificial intelligence algorithm. Dynamically short-term and long-term rewards are provided for energy and computational cost expenditures.

Table 3 presents a comparison of the different resource allocation approaches. To summarize, the majority of research is based on considering resource allocation problems, such as resource optimization and middleware framework. In the first approach, many researchers have developed mathematical models using the graph theory, relying on operation research applying different meta-heuristic approaches, such as ant colony optimization, particle swarm optimization and

Table 4

Major metrics used in resource allocation approaches

Metrics	References
Network lifetime	[38], [53]
Throughput	[38], [40], [42], [59]
Data traffic	[39], [48]
Service rate	[42], [56], [58]
Latency	[46], [51], [53]
Error rate	[47], [49]
Device utilization	[40], [42], [44], [45], [46], [49], [58], [61]
Computation cost	[41], [43], [47], [48], [54], [57]
Service rate	[42]
Node lifetime	[45], [51], [52], [55], [56], [59], [60], [61]

Table 5

Architecture types used in resource allocation

Metrics	References
Two-tier architecture	[38], [39], [41], [42], [43], [45], [48], [49], [50], [52], [54], [55], [56], [57], [59], [60]
Three-tier architecture	[40], [44], [46], [47], [51], [53], [58], [61]

genetic algorithm. Many authors also use different game theory approaches, such as cooperative game, evolutionary game and Stackelberg game, where interactions between the data service provider, access point, fog and IoT devices are modeled as conflicting games. In the other approach, most of the work is based on developing a middleware prototype for resource management purposes. Only a few works focus on testing resource allocation strategies using the standard framework. But testing with the use of the real framework is really important to understand the behavior of the proposed algorithm and its assumptions in a real world scenario. In Table 4, the important resource allocation metrics are presented in terms of scalability, reduced communication and computation cost and load balancing aiming to improve node and network lifetime. Edge and fog computing are promising solutions for highly computationally-intensive operations that delay sensitive applications. Table 5 compares the number of resource allocation schemes proposed for 3-tier and 2-tier architectures.

### 3. IoT Resource Management Module in FIWARE

To manage diverse technologies and components of IoT, generic enablers (GEs) or abstract interfaces have been developed. For example, the European ICT and standardization committee proposes an open-source middleware platform known as FIWARE [64]. Another popular open-source project – IoTivity – proposed by the Open Connec-

tivity Foundation (OCF) [65], also outlines similar responsibilities of the resource management module. Both architectures incorporate resource identification and addressing schemes, mapping of device ID to a specific device, dynamic linking of resources, as well as mechanisms for controlling access to resources and information in the resource management module. They also include device management functions, such as activation, deactivation, software update or remote monitoring. The FIWARE architecture proposes four GEs, such as communication, resource management, data handling and process automation. The resource management process is shown in Fig. 8.

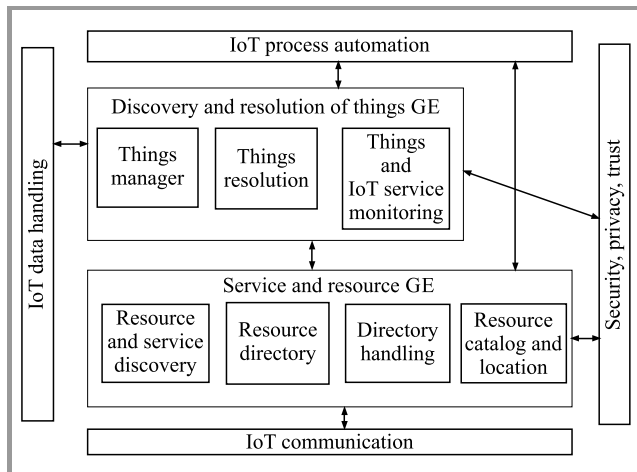


Fig. 8. Resource management using a generic enabler (GE).

This abstraction layer provides unified service and operation support for the management of IoT resources. A resource management GE is used and is responsible for identifying, registering, unregistering, discovering, activating, utilizing and configuring resources. It consists of such sub-layers as discovery, resolution of things, as well as services and resources interaction. These are described as follows:

- **Discovery and resolution of things generic enablers.** They help discover services and resources based on the device ID.
- **Services and resources interaction generic enablers.** These provide such services as registration, deregistration, status retrieval and updates performed by sub-modules. Before providing a given service, the enabler interacts with the IoT process automation module for context and mobility information. It also interacts with the security and privacy module to verify access permission for a given request.

#### 4. Challenges and Research Issues

Each year a great number of smart objects is integrated with the IoT ecosystem. In the view of the scale and heterogeneity of the phenomenon, there is a need for dynamic, scalable and robust solutions allowing to manage IoT resources. After an extensive survey of existing solutions,

we have identified the following major issues related to resource management and requiring further research to be performed:

- Interoperability of IoT resources used on different platforms – there is no common description for resources, their capabilities and format of data exchanged between different SDOs. Interoperability across multiple SDOs simplifies the resource management process across multiple domains;
- Prioritization of discovered resources helps the resource management module select the appropriate resource;
- Security – standard security mechanism for resource discovery;
- Load balancing – for optimal utilization of resources, a distributed resource management module is required that is capable of relying on the benefits offered by fog and edge computing schemes;
- Dynamic task distribution mechanisms – based on the capabilities of the participating devices. There is great room for novel dynamic task management modules based on the resources available within the system;
- Resource estimation – to manage the resources effectively;
- Predictive mechanisms/algorithms – used in resource allocation;
- Resource scheduling and allocation – for real time IoT applications;
- Improved pricing model – for resources used in cloud, fog, edge and for IoT;
- Fault tolerance – to provide uninterrupted services to the user during node failure. Most of the proposed analytical models fail to accurately map actual resource allocation problems experienced in IoT. Also, many assumptions made while modeling the problem fail to address the major challenges, just as the majority of research does not discuss ways to integrate the local optimal solution identified with global solutions of resource allocation-related problems.

No unified standard exists accepted across the entire market. Individual SDOs and industry alliances have defined their own formats. This prevents end users from being able to take advantage of improved services. To ensure cooperation between SDOs, SSOs and industry alliances and to identify gaps between them, special groups – such as the ESTI specialist task group [66], the alliance of Internet of Things innovations (AIOTTI) [67] or CREATE-IOT [68] have been formed. Even though work towards standardization is already in progress, adaptation of the results on the

market is still very low. To address these issues, there is an urgent need for research to develop a dynamic and efficient broker capable of understanding and mapping multiple standard resources. Also, there is a need to provide software developers with fine-grained descriptions in order to avoid ambiguity caused by abstract information from SDOs.

The management of IoT resources used in each layer of the environment poses a number of challenges. Most work is performed to develop mathematical resource allocation models. But these proposed models are tested only with the use of simulation tools or software developed in-house. So, there is an immediate need for framework or test bed based verification of algorithms.

Semantic web-based modeling, although enhancing interoperability, also suffers from incompatibility between multiple SDOs, SSOs and industrial alliances. Semantic-based technologies experience a range of other issues, such as the need for lightweight semantic technology, improved security mechanism for registration and resource discovery, large indexing and querying capacity, dynamic conversion between standards, efficient streaming models and open standards or interfaces for application developers.

## 5. Conclusion

In this paper, heterogeneity of IoT devices, distributed architectures of IoT ecosystems, types of resources, challenges in managing the heterogeneous resources, phases of resource management in IoT, guidelines provided by SDOs for interoperability in managing resources and potential design options available in each resource management phase (i.e. semantic- or virtualization-based modeling, centralized or distributed resource discovery) are discussed.

The key outcome of this study indicates that most algorithms discussed in the resource allocation section are mathematical or attribute-based. Assumptions made and results obtained with the use of these algorithms are hard to verify in real frameworks or testbeds. Our immediate research will focus on proposing a resource allocation algorithm considering the distributed architecture of the IoT ecosystem. The proposed algorithm will be verified using the existing IoT framework. Furthermore, open issues related to resource management schemes, as discussed in this paper, are expected to motivate researchers to conduct further investigations in this particular area.

## References

- [1] A. Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications", *IEEE Communi. Surv. Tutor.*, vol. 17, no. 4, pp. 2347–2376, 2015 (DOI: 10.1109/COMST.2015.2444095).
- [2] M. Burhan, R. A. Rehman, B. Khan, and B. Kim, "IoT elements, layered architectures and security issues: a comprehensive survey", *J. of Sensors*, vol. 18, no. 9, pp. 2796– 831, 2018 (DOI: 10.3390/s18092796).
- [3] "IoT Reference Model" [Online]. Available: <https://www.itu.int/ITU-T/recommendations/rec.aspx?rec=y.2060>
- [4] F. Delicato, P. Pires, and T. Batista, *Resource Management for Internet of Things*. Springer, 2017 (ISBN: 9783319542478).
- [5] Z. Ghanbari, N. J. Navimipour, M. Hosseinzadeh, and A. Darwesh, "Resource allocation mechanisms and approaches on the Internet of Things", *J. of Cluster Comput.*, vol. 22, no. 4, pp. 1253–1282, 2019 (DOI: 0.1007/s10586-019-02910-8).
- [6] A. Botta, W. Donato, V. Persico, and A. Pescapé, "Integration of cloud computing and Internet of Things", *J. of Fut. Gener. Com. Syst.*, vol. 56, pp. 684–700, 2016 (DOI: 10.1016/j.future.2015.09.021).
- [7] "ITU Common Functional and Non Functional requirement of IoT" [Online]. Available: <https://www.itu.int/rec/T-REC-Y.2066>
- [8] M. Glinz, "On non-functional requirements", in *Proc. of 15th IEEE Int. Requir. Engin. Conf.*, Delhi, India, 2007, pp. 21–22 (DOI: 10.1109/RE.2007.45).
- [9] A. N. Lam and O. Haugen, "Applying semantics into service-oriented IoT framework", in *Proc. of IEEE 17th Int. Conf. on Indust. Inform. INDIN 2019*, Helsinki, Finland, 2019, pp. 206–213 (DOI: 10.1109/INDIN41052.2019.8972295).
- [10] A. Sheth, C. Henson, and S. Sahoo, "Semantic sensor web", *IEEE J. on Internet Comput.*, vol. 12, no. 4, pp. 1–10, 2008 (DOI: 10.1109/MIC.2008.87).
- [11] P. Barnaghi, M. Presser, and K. Moessner, "Published linked sensor data", in *Proc. of Int. Symp. on Collabor. Technol. and Syst.*, pp. 1–16, 2010, in ISWC 2010, 2010-11-07–2010-11-11, Shanghai, China [Online]. Available: <http://eprints.surrey.ac.uk/470673/1/sense2web.pdf>
- [12] I. Alam *et al.*, "IoT virtualization: A survey of software definition and function virtualization techniques for Internet of Things", pp. 1–30, 2019 [Online]. Available: <https://arxiv.org/pdf/1902.10910.pdf>
- [13] B. I. Ismail, "Evaluation of docker as edge computing platform", in *Proc. of IEEE Int. Conf. on Open Syst.*, Bandar Melaka, Malaysia, 2015, pp. 130–135 (DOI: 10.1109/ICOS.2015.7377291).
- [14] J. G. Ko *et al.*, "Sensor virtualization module: virtualizing IoT devices on mobile smart phones for effective sensor data management", *J. of Distrib. Sensor Netw.*, vol. 11, no. 3, pp. 1–10, 2015 (DOI: 10.1155/2015/730762).
- [15] B. Billet and V. Issarny, "From task graphs to concrete actions: A new task mapping algorithm for the future Internet of Things", in *Proc. of IEEE 11th Int. Conf. on Mob. ad hoc and Sensor Syst.*, Philadelphia, PA, USA, 2014, pp. 470–478 (DOI: 10.1109/MASS.2014.20).
- [16] V. Angelakis, I. Avgouleas, N. Pappas, E. Fitzgerald, and D. Yuan, "Allocation of Heterogeneous resources of an IoT device to flexible services in IoT", *J. on IEEE Internet of Things*, vol. 3, no. 5, pp. 691–700, 2016 (DOI: 10.1109/JIOT.2016.2535163).
- [17] M. Sharief, O. Kingston, S. Hossam, and O. Kingston, "Resource re-use in wireless sensor networks: realizing a synergetic Internet of Things", *J. of Commun.*, vol. 7, no. 7, pp. 484–493, 2012 (DOI: 10.4304/jcm.7.7.484-493).
- [18] P. Banerjee *et al.*, "Everything as a service: powering the new information economy", *J. of Comp.*, vol. 44, no. 3, pp. 36–43, 2011 (DOI: 10.1109/MC.2011.67).
- [19] W. Wang, S. De, G. Cassar, and K. Moessner, "Knowledge representation in the Internet of Things: semantic modeling and its applications", *Automatika J. for Control, Measur., Electron., Comput.*, vol. 54, no.4, pp. 388–400, 2013 (DOI: 10.7305/automatika.54-4.414).
- [20] J. Soldatos *et al.*, "OpenIoT: open source Internet-of-Things in the cloud", in *Interoperability and Open-Source Solutions for the Internet of Things*, LNCS, vol. 9001, pp. 13–25. Springer, 2015 (DOI: 10.1007/978-3-319-16546-2.3).
- [21] M. Compton *et al.*, "The SSN Ontology of the W3C Semantic Sensor Network", *J. of Web Semant.*, vol. 17, pp. 25–32, 2012 (DOI: 10.1016/j.websem.2012.05.003).
- [22] S. Nagowah and B. A. Rahimbux, "An overview of semantic interoperability ontologies and frameworks for IoT", in *Proc. of IEEE 6th Int. Conf. on Enterpr. Syst. ES 2018*, Limassol, Cyprus, 2018, pp. 470–478 (DOI: 10.1109/ES.2018.00020).

- [23] A. Palade, C. Cabrera, G. White, M. A., Razzaque, and S. Clarke, "Middleware for Internet of Things: A quantitative evaluation in small scale", in *Proc. of IEEE 18th Int. Symp. on A World of Wirel., Mob. and Multim. Netw. WoWMoM 2017*, Macau, China, 2017, pp. 1–6 (DOI: 10.1109/WoWMoM.2017.7974340).
- [24] K. Ogawa, K. Kanai, K. Nakamura, H. Kanemitsu, J. Katto, and H. Nakazato, "IoT device virtualization for efficient resource utilization in smart city IoT platform", in *Proc. of IEEE Int. Conf. on Perv. Comput. and Commun. Worksh. PerCom Workshops 2019*, Kyoto, Japan, 2019, pp. 419–422 (DOI: 10.1109/PERCOMW.2019.8730806).
- [25] S. K. Datta, R. P. Costa, and C. Bonnet, "Resource discovery in Internet of Things: Current trends and future standardization aspects", in *Proc. of IEEE 2nd World Forum on Internet of Things WF-IoT 2015*, Milan, Italy, 2015, pp. 542–547 (DOI: 10.1109/WF-IoT.2015.7389112).
- [26] N. Hussain, T. Anees, and AzeemUllah, "Development of a novel approach to search resources in IoT", *Int. J. of Adv. Comp. Sci. and Appl.*, vol. 9, no. 9, pp. 385–398, 2018 (DOI: 10.14569/IJACSA.2018.090949).
- [27] C. Perera and A. V. Vasilakos, "A knowledge-based resource discovery for Internet of Things", *J. of Knowledge-Based Syst.*, vol. 109, pp. 122–136, 2016 (DOI: 10.1016/j.knosys.2016.06.030).
- [28] M. Afrin and R. Mahmud, "Software defined network-based scalable resource discovery for Internet of Things", *J. of EAI Endorsed Trans. on Scalable Inform. Syst.*, vol. 17, no. 14, pp. 1–6, 2017 [Online]. Available: <https://eudl.eu/pdf/10.4108/eai.25-9-2017.153149>
- [29] P. Krivic, P. Skocir, and M. Kusek, "Agent-based approach for energy-efficient IoT services discovery and management", in *Agents and Multi-Agent Systems: Technologies and Applications 2018 Proceedings of the 12th International Conference on Agents and Multi-Agent Systems: Technologies and Applications (KES-AMSTA-18)*. Springer, 2018, pp. 57–66 (DOI: 10.1007/978-3-319-92031-3.6).
- [30] L. Nunes, J. Estrella, C. Perera, S. Reiff-Marganiec, and A. Delbem, "Multi-criteria IoT resource discovery: a comparative analysis", *J. on Software-Practice and Exper.*, vol. 47, no. 10, pp. 1325–1341, 2016 (DOI: 10.1002/spe.2469).
- [31] K. Khalil, K. Elgazzar, and M. Bayoumi, "A comparative analysis on resource discovery protocols for the Internet of Things", in *Proc. of IEEE Global Commun. Conf. GLOBECOM-2018*, Abu Dhabi, United Arab Emirates, 2018 (DOI: 10.1109/GLOBECOM.2018.8647553).
- [32] M. Aazam, M. St-Hilaire, C. Lung, I. Lambadaris, and E. Huh, "IoT resource estimation challenges and modeling in fog", in *Fog Computing in the Internet of Things*. Springer, 2018, pp. 17–31 (DOI: 10.1007/978-3-319-57639-8.2).
- [33] M. Aazam and E. N. Huh, "Fog computing micro data center based dynamic resource estimation and pricing model for IoT", in *Proc. of the 29th IEEE Int. Conf. on Adv. Inform. Netw. and Appl.*, Gwangju, South Korea, 2015, pp. 687–694 (DOI: 10.1109/AINA.2015.254).
- [34] M. Aazam and E. Huh, "Resource management in media Cloud of Things", in *Proc. of IEEE Int. Conf. on Parall. Process. Worksh.*, Minneapolis, MN, USA, 2014, pp. 361–367 (DOI: 10.1109/ICPPW.2014.54).
- [35] M. Aazam and E. Huh, "Dynamic resource provisioning through fog micro data center", in *Proc. of the IEEE Int. Conf. on Perv. Comput. Commun. Worksh. PerCom Workshops 2015*, St. Louis, MO, USA, 2015, pp. 105–110 (DOI: 10.1109/PERCOMW.2015.7134002).
- [36] M. Aazam, M. St-Hilaire, C. Lung, and I. Lambadaris, "PRE-Fog: IoT trace based probabilistic resource estimation at fog", in *Proc. of the 13th IEEE Ann. Consumer Commun. and Netw. Conf. CCNC 2016*, Las Vegas, NV, USA, 2016, pp. 12–17 (DOI: 10.1109/CCNC.2016.7444724).
- [37] E. Solaiman, R. Ranjan, P. Jayaraman, and K. Mitra, "Monitoring Internet of Things application ecosystems for failure", *J. on IEEE IT Profess.*, vol. 18, no. 5, pp. 1–4, 2016 (DOI: 10.1109/MITP.2016.90).
- [38] X. Liu, Z. Qin, Y. Gao, and J. A. McCann, "Resource allocation in wireless powered IoT networks", *IEEE Internet of Things J.*, vol. 6, no. 3, pp. 4935–4945, 2019 (DOI: 10.1109/JIOT.2019.2895417).
- [39] S. Manakkadu and S. Dutta, "On efficient resource allocation in the Internet of Things environment", in *Proc. of 8th Int. Conf. on the Internet of Things*, Santa Barbara, CA, USA, 2018, no. 22, pp. 1–5 (DOI: 10.1145/3277593.3277623).
- [40] S. F. Abedin *et al.*, "Resource allocation for ultra-reliable and enhanced mobile broadband IoT applications in fog network", *J. on IEEE Trans. on Commun.*, vol. 67, no. 1, pp. 489–502, 2019 (DOI: 10.1109/TCOMM.2018.2870888).
- [41] C. Tsai and S. Liu, "An effective IoT service-to-interface assignment algorithm via search economics", *J. on IEEE Internet of Things*, vol. 5, no. 3, pp. 1708–1718, 2018 (DOI: 10.1109/JIOT.2018.2796310).
- [42] S. Kim, "Asymptotic shapley value based resource allocation scheme for IoT services", *J. on Comp. Netw.*, vol. 100, pp. 55–63, 2016 (DOI: 10.1016/j.comnet.2016.02.021).
- [43] C. Tsai, "SEIRA: An effective algorithm for IoT resource allocation problem", *J. on Comp. Commun.*, vol. 119, pp. 156–166, 2017 (DOI: 10.1016/j.comcom.2017.10.006).
- [44] H. Zhang, Y. Xiao, S. Bu, D. Niyato, F. R. Yu, and Z. Han, "Computing resource allocation in three-tier IoT fog networks: A joint optimization approach combining Stackelberg game and matching", *J. on IEEE Internet of Things*, vol. 4, no. 5, pp. 1204–1215, 2017 (DOI: 10.1109/JIOT.2017.2688925).
- [45] V. Pilloni and L. Atzori, "Consensus-based resource allocation among objects in the Internet of Things", *J. on Annals of Telecommun.*, vol. 72, pp. 415–429, 2017 (DOI: 10.1007/s12243-017-0583-6).
- [46] A. Nassar and Y. Yilmaz, "Resource allocation in fog RAN for heterogeneous IoT environments based on reinforcement learning", in *Proc. of IEEE Int. Conf. on Commun. ICC 2019*, Shanghai, China, 2019, pp. 1–6 (DOI: 10.1109/ICC.2019.8761626).
- [47] G. Colistra, V. Pilloni, and L. Atzori, "The problem of task allocation in the Internet of Things and the consensus-based approach", *J. of Comp. Netw.*, vol. 73, pp. 98–111, 2014 (DOI: 10.1016/j.comnet.2014.07.011).
- [48] M. Kim and I. Ko, "An efficient resource allocation approach based on a genetic algorithm for composite services in IoT environments", in *Proc. of IEEE Int. Conf. on Web Services*, New York, NY, USA, 2015, pp. 543–550 (DOI: 10.1109/ICWS.2015.78).
- [49] H. Lan, H. Song, H. Liu, and G. Y. Zhang, "Heterogeneous-oriented resource allocation method in Internet of Things", *J. of Appl. Mechan. and Mater.*, vol. 427, pp. 2791–2794, 2013 (DOI: 10.4028/www.scientific.net/AMM.427-429.2791).
- [50] L. Zheng, L. Kaihua, S. Yuting, and M. Yongtao, "Adaptive resource allocation algorithm for Internet of Things with bandwidth constraint", *J. on Trans. of Tianjin Univer.*, vol. 18, pp. 253–258, 2012 (DOI: 10.1007/s12209-012-1873-8).
- [51] A. Kliem and O. Kao, "The Internet of Things resource management challenge", in *Proc. of IEEE Int. Conf. on Data Sci. and Data Intensive Syst.*, Sydney, Australia, 2015, pp. 483–490 (DOI: 10.1109/DS-DIS.2015.21).
- [52] T. Renner, A. Kliem, and O. Kao, "The device cloud – applying cloud computing concepts to the Internet of Things", in *Proc. of IEEE 11th Int. Conf. on Ubiquit. Intell. and Comput., IEEE 11th Int. Conf. on Autonom. and Trust. Comput. and IEEE 14th Int. Conf. on Scal. Compu. and Commun. and Its Assoc. Worksh.*, Bali, Indonesia, 2014, pp. 396–401 (DOI: 10.1109/UIC-ATC-ScalCom.2014.106).
- [53] Q. Wang, and S. Chen, "Latency-minimum offloading decision and resource allocation for fog-enabled Internet of Things networks", *J. of Trans. on Emerg. Telecommun. Technol.*, 2020 [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.3880> (DOI: 10.1002/ett.3880).
- [54] X. Liu, Y. Li, X. Zhang, W. Lu, and M. Xiong, "Energy-efficient resource optimization in green cognitive Internet of Things", *J. of Mob. and Netw. Appl.*, 2020 (DOI: 10.1007/s11036-020-01510-w).
- [55] Q. Fan and N. Ansari, "Application aware workload allocation for edge computing-based IoT", *J. on IEEE Internet of Things*, vol. 5, no. 3, pp. 2146–2153, 2018 (DOI: 10.1109/JIOT.2018.2826006).



- [56] W. Ejaz and M. Ibnkahla, "Multiband spectrum sensing and resource allocation for IoT in cognitive 5G networks", *J. of IEEE Internet of Things*, vol. 5, no. 1, pp. 150–163, 2018 (DOI: 10.1109/JIOT.2017.2775959).
- [57] A. Rullo, D. Midi, E. Serra, and E. Bertino, "Pareto optimal security resource allocation for Internet of Things", *J. on ACM Trans. on Priv. and Secur.*, vol. 20, no. 4, pp. 1–30, 2017 (DOI: 10.1145/3139293).
- [58] Z. Yu, Y. Gong, S. Gong, and Y. Guo, "Joint task offloading and resource allocation in UAV-enabled mobile edge computing", *J. on IEEE Internet of Things*, vol. 7, no. 4, pp. 3147–3159, 2020 (DOI: 10.1109/JIOT.2020.2965898).
- [59] Y. Jiao, P. Wang, D. Niyato, and Z. Xiong, "Social welfare maximization auction in edge computing resource allocation for mobile blockchain", in *Proc. of IEEE Int. Conf. on Commun. ICC 2018*, Kansas City, MO, USA, 2018 (DOI: 10.1109/ICC.2018.8422632).
- [60] X. Xu *et al.*, "Dynamic resource allocation for load balancing in fog environment", *J. on Wirel. Commun. and Mob. Comput.*, vol. 2, 2018 (DOI: 10.1155/2018/6421607).
- [61] H. Malik *et al.*, "Radio resource management scheme in NB-IoT systems", *J. of IEEE Access*, vol. 6, pp. 15051–15064, 2018 (DOI: 10.1109/ACCESS.2018.2812299).
- [62] Z. Dou, G. Si, Y. Lin, and M. Wang, "An adaptive resource allocation model with anti-jamming in IoT network", *J. of IEEE Access*, vol. 7, pp. 93250–93258, 2019 (DOI: 10.1109/ACCESS.2019.2903207).
- [63] F. Xu, F. Yang, S. Bao, and C. Zhao, "DQN inspired joint computing and caching resource allocation approach for software defined information-centric Internet of Things network", *J. of IEEE Access*, vol. 7, pp. 61987–61996, 2019 (DOI: 10.1109/ACCESS.2019.2916178).
- [64] FIWARE Open source Platform [Online]. Available: <https://www.fiware.org/>
- [65] IoTivity Open source Platform [Online]. Available: <https://iotivity.org/>
- [66] European Special Task Force 505 for IoT standards landscaping and IoT European Large Scale Pilot gap analysis [Online]. Available: <https://portal.etsi.org/STF/STFs/STFHomePages/STF505>
- [67] Standardization Gaps by The Alliance for Internet of Things Innovation [Online]. Available: <https://aioti.eu/aioti-wg03-reports-on-iot-standards/>
- [68] European Large Scale Pilot Programme [Online]. Available: <https://european-iot-pilots.eu/>



**Lokesh B. Bhjantri** received his Ph.D. degree in Computer Science and Engineering from the Visveswaraiiah Technological University (VTU), Belgaum, Karnataka, in 2015. He has been working, for 15 years now, as an Associate Professor at the Department of Information Science and Engineering,

Basaveshwar Engineering College, Bagalkot, India. His areas of interest include distributed/wireless sensor networks, cognitive Internet of Things, mobile computing and communications, networking protocols, genetic algorithms, applications of agents, and real time systems. He is a member of the Board of Computer Science, Engineering and Information Science Studies. He is also a member of IEEE, International Association of Computer Science and Information Technology (IACSIT) and International Association of Engineers (IAENG). He has given invited lectures at AICTE and TEQIP sponsored workshops and under faculty development programs, as a supporting speaker.

 <https://orcid.org/0000-0002-3947-4292>

E-mail: [lokeshcse@yahoo.co.in](mailto:lokeshcse@yahoo.co.in)

Department of Computer Science and Engineering  
Basaveshwar Engineering College (Autonomous)  
Bagalkot, Karnataka, India



**Gangadharaiiah S.** received the B.Eng. and M.Tech. degree in Computer Science and Engineering from Visvesvaraya Technological University, Belagavi, India, in 2003 and 2011, respectively. He is currently pursuing a Ph.D. degree in Computer Science and Engineering at Visvesvaraya Technological University, Belagavi,

India. He has been working as an Assistant Professor at the Department of Computer Science at S.R.S.M.N. Government First Grade College at Udupi, India, since 2017. He worked as a junior Design Engineer and Software Engineer for Federal Technologies and Mindtree Pvt. Ltd. from 2004–2008, respectively. He also worked as an Assistant Professor at the Department of Information Science and Engineering at Acharya Institute of Technology, Bangalore, from 2009 to 2017. His current research interests include Internet of Things, sensor networks, and cloud computing.

 <https://orcid.org/0000-0001-9233-7740>

E-mail: [gangadhar.s@gmail.com](mailto:gangadhar.s@gmail.com)

Department of Computer Science  
S.R.S.M.N. Government First Grade College  
Udupi, Karnataka, India