Review

A survey study on Internet of Things resource management

Abishi Chowdhury*, Shital A. Raut

Department of Computer Science and Engineering, Visvesvaraya National Institute of Technology, Nagpur, Maharashtra 440010, India

ARTICLE INFO

ABSTRACT

In the modern era of computation, Internet of Things (IoT) has attracted more and more attention for implementing automation based high quality eco-system, innovative and reliable services, and improved productivity. That is why, IoT is presently spanning over a diverse range of applications in the fields of education, healthcare, agriculture, military, industry, etc. The resources IoT requires in those applications are heterogeneous in nature, for example, storage, processor, network, sensor nodes, energy, etc. The obvious challenge is how to manage these limited resources efficiently. Quite recently a number of attempts have been made towards an optimal resource management using various scheduling schemes. This paper broadly divides these scheduling schemes into three aspects, namely Quality of Services (QoS), IoT architecture, and IoT network infrastructure. The paper also presents an analysis based on different metrics of IoT resource scheduling and a comparison among different resource scheduling techniques. A section of this analysis is dedicated to the simulation model analyses involved in different scheduling schemes. Finally, we believe there is still a lot of scope in improving IoT resource management maintaining the overall quality needed. The paper considers with such a future direction of research in this field.

1. Introduction

Internet of Things (IoT) aims to bring the objects of our day to day life under a network and make them programmable with the help of some eminent technologies like Radio Frequency Identification (RFID), Ubiquitous connectivity, and sensor network technologies (Ashton, 2009; Rowland et al., 2015; Whitmore et al., 2015). Thus, IoT intends to achieve pervasive M2M (machine to machine) connectivity and to provide secure, real time services such as remote monitoring, sensing, tracking, locating, scheduling, controlling, planning, maintenance, data mining, decision support, and others for a connected world that is highly scalable, energy efficient, productive and environmental friendly (Zhou, 2013). So, basically IoT is a complex distributed eco-system where the things are connected with each other and working together towards a goal. The complexity of IoT eco-system can be understood considering the heterogeneous and dynamic nature of resources associated with the numerous nodes of this eco-system.

The physical resources of an IoT eco-system are the nodes connected together to form that particular eco-system; each of these physical resources/devices is assumed to have one or more internal resources which can be identified by a Uniform Resource Identifier (URI) (Rahman and Wang, 2016). These resources can be the computational resources, storage capacity, and energy resources. Another type of IoT

resource is the resources corresponding to the communication channel or the network resources. Fig. 1 shows the different types of IoT resources that are used in several scheduling processes in IoT resource management.

Resource management is one of the most important tasks in distributed systems and has been a subject of research for years. An efficient resource management involves discovering and identifying all available resources, selecting appropriate resources, and partitioning and provisioning them to optimize the utility function which can be in terms of overall performance, cost, energy efficiency, information accuracy, coverage, reliability, etc. (Buyya and Dastjerdi, 2016). Several potential research regarding resource management in different computing areas have already been exhaustively carried out. Still, IoT resource management is an open challenge for the researchers that call for innovative solutions. The following prime reasons make the task more onerous:

· heterogeneity of various types of IoT resources
· dynamic nature of IoT eco-systems and system resources
· resource constraint IoT nodes
· failures of diverse IoT eco-system resources can lead to performance degradation

* Corresponding author.
  E-mail addresses: abishi.chowdhury@gmail.com (A. Chowdhury), saraut@cse.vnit.ac.in (S.A. Raut).
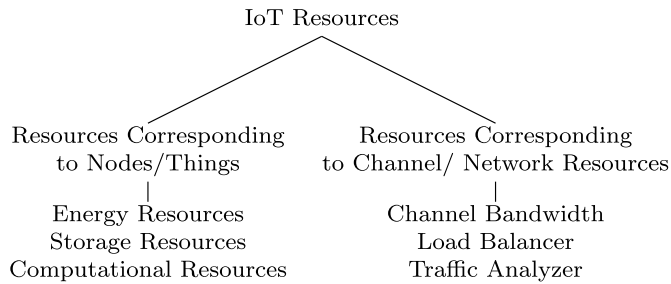
Fig. 1. Classification of IoT resources.

To overcome these aforementioned challenges, scheduling the tasks (may be an IoT user request that can be a combination of several other tasks or some internal task of an IoT eco-system) on appropriate available physical resources becomes necessary. Sometimes, the real time applications of IoT need to follow a strict deadline; therefore, it is the responsibility of the scheduler to manage the IoT resources in such way that scheduling process can meet the deadlines. Fig. 2 shows the taxonomy of entire IoT resource management activities. Resource or service discovery is the *first step* of resource management which targets to identify and locate the actual device and then discover the objective service that needs to be invoked. The *second step* for satisfying resource provisioning requirements in IoT is to efficiently partition the resources and obtain a higher utilization rate. This concept is immensely used in another distributed computing system i.e. cloud computing through virtualization techniques and commodity infrastructures.

Inadequacy of an efficient resource discovery algorithm can result in runtime failures, execution delays, energy outage, and poor user experience. An effective scheduling algorithm (Liu et al., 2014a) that dynamically selects centralized or flooding strategies can reduce the consumed energy which is the primary resource of IoT, even though some other parameters for example latency and mobility should be shared in to provide a convenient solution for IoT, considering its dynamic nature.

1.1. Motivation for research: the challenges

One of the biggest challenges in the area of parallel and distributed computing paradigm is the competence to manage heterogeneous types of tasks while ensuring the Quality of Services, Automatic Management

of resources, different types of Architecture and Network Infrastructure requirements.

· Resource scheduling in IoT complex eco-system is a process of dynamic allocation of IoT workloads after effective resource/service discovery and resource provisioning process. This study points out the different available resource scheduling algorithms based on specific scheduling criteria.
· We recognize the importance of methodological survey on IoT resource scheduling as so far from the best of our belief there is no comprehensive survey available regarding this topic. Therefore, we have listed down some of the major scheduling techniques, available simulators, and relevant research directions in this area with the expectation that this can be a brief manual to understand the area for the new comers in this field.

1.2. Our contributions

In this review, we firstly realize the importance of resource scheduling and the overall scheduling process in an IoT environment. Our main focus is on the analysis of different scheduling techniques which can be used to fetch the full leverage of IoT resources. Some of the key features of this analysis are:

· Figure out the essence of IoT resource scheduling and the issues and challenges that need to be tackled during resource provisioning and scheduling.
· Illustrate the overall IoT resource scheduling phenomenon.
· There can be different parameters for IoT due its vast application domains. Identification of such parameters that are used for IoT resource scheduling.
· Resource scheduling considering target QoS.
· Architectural based different scheduling techniques.
· Underlying network based scheduling techniques.
· Comparison among various scheduling techniques and listing different simulators used for evaluation of these techniques.
· Some emerging future research trends regarding IoT resource scheduling.

The rest of the paper is organized as section 2 elaborates the significance of IoT resource scheduling. Section 3 shows the concept of IoT resource scheduling process. Section 4 describes different aspects of IoT
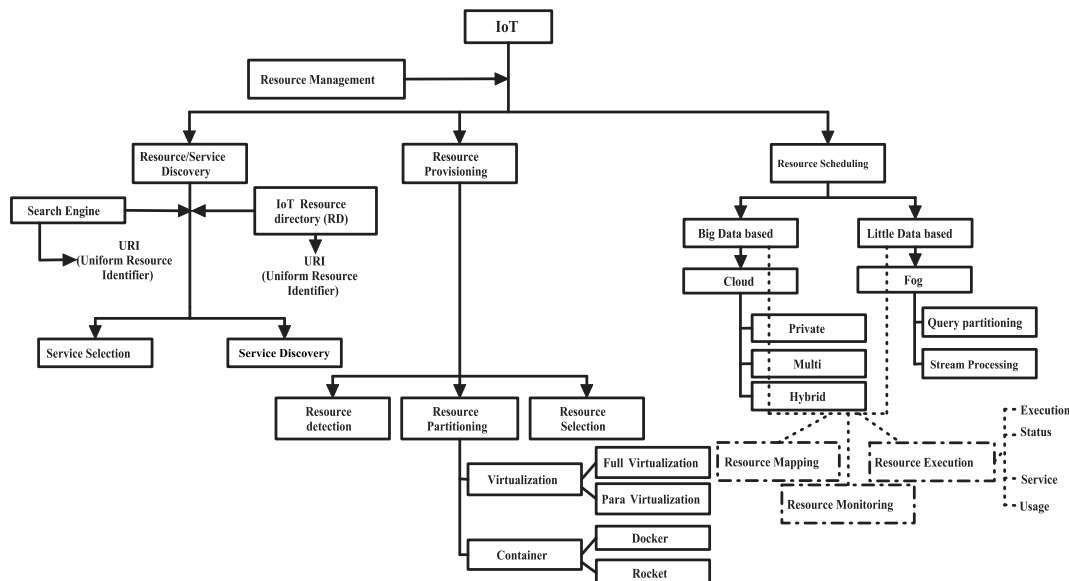

Fig. 2. IoT resource management activities.

resource scheduling and under this, section 4.1 explains the QoS aspects of IoT and several scheduling techniques based on QoS requirements, section 4.2 shows the scheduling schemes based on different architectures, some network infrastructure based scheduling schemes have been described in section 4.3. Section 5 discusses the analysis part; under this section 5.1 describes different metrics based analysis of IoT resource scheduling, section 5.2 shows a comparison of different IoT resource scheduling techniques, and 5.3 presents the simulation model analysis. Section 6 contributes different research needs regarding resource scheduling in IoT and finally, section 7 concludes the paper.

2. Significance of resource scheduling in Internet of Things (IoT): the issues and challenges

Resource scheduling in IoT indicates the process of coordination between different IoT resources that the IoT services access in the IoT environment. The scheduler needs to keep the track of IoT resources that are used to serve different IoT services (Buyya and Dastjerdi, 2016). Tracking of resources is one of the prerequisites for the implementation of resource optimization techniques at the sensors/IoT levels (eg. Data streaming according to IoT application needs). Effective scheduling of these resources is no doubt a great challenge and it has to deal with several issues. A few of these has been listed below:

· Heterogeneous tasks, and heterogeneous and dynamic nature of resources in IoT
· Resource constraint IoT nodes: mainly, energy resource, storage resource, and processing power
· Dynamic and real-time applications of IoT
· Services based scheduling, priority estimation of services and scheduling on that basis
· Reliability issues for services and nodes of IoT
· Channel interference in the case of channel based scheduling

While handling these issues, the scheduling process must take care of the metrics of scheduling which need to be either minimized or maximized (see Table 1).

In this context, before starting any scheduling process some questions or research challenges need to be addressed and as a solution of one or more challenging questions, scheduling of resources occurs. Some of the important research questions that result in resource scheduling in IoT environment are: How to handle coexistence issues in sensor networks? How to handle co-channel interference? How to respond a user request for heterogeneous devices? How to manage the

Table 1
Objectives of IoT resource scheduling.

Metrics	Need to be
Throughput (T)	Maximize
Response Time (RT)	Minimize
Latency (LT)	Minimize
Fairness (F)	Maximize
Jitter (J)	Minimize
Traffic (TF)	Minimize
Average Waiting Time (AWT)	Minimize
Delay (D)	Minimize
Load (L)	Minimize (Balance)
Reliability (R)	Maximize
Information Accuracy (IA)	Maximize
Coverage of IoT (Cov)	Maximize
Energy Consumption (EC)	Minimize
Network Life (NL)	Maximize
Blocking Probability (BP)	Minimize
Co-channel Interference (CCI)	Minimize
Service Cost (SC)	Minimize
Service Time (ST)	Minimize
Resource Utilization (RU)	Maximize

power consumption in IoT devices? How to handle spatial request for IoT devices? How to handle M2M traffic? How to allocate cloud resources in an IoT environment? How to manage virtual machines for heterogeneous task allocation? How to manage the load in an IoT device? How to reduce the waiting time for a service? How to reduce the communication cost? How to reduce the packet loss rate? How to reduce the blocking probability of incoming requests? How to increase the throughput in an IoT network? How to ensure the accuracy level of generated information? How to manage resources in heterogeneous network architecture? How to develop a single architecture that can satisfy QoS requirements of IoT users? How to get the maximum utilization of the IoT resources?

Now, a question that can arise here is, though there are already several state-of-the-art algorithms available in the two enabling technologies (directly correlated with IoT) (Lin et al., 2017) of IoT, i.e. cloud computing as well as Wireless Sensor Networks (WSNs) for resource management. So, why not these techniques can be used for IoT resource management as well? The answer is that the traditional server scheduling techniques in cloud computing cannot be able to serve IoT as it comprises innumerable heterogeneous devices and real time applications which are lacking in standardization. Therefore, to satisfy IoT users' expectations, the conventional cloud scheduling techniques should be upgraded so that it can efficiently schedule and allocate resources to the IoT requests (Narman et al., 2016). On the other hand, WSN, consists of different homogeneous and heterogeneous sensors, aims to gather data from these sensors of the subject environment. To answer the need of IoT applications, WSNs have been upgraded to form IoT network through technological evolutions (Castellani et al., 2010; Mainetti et al., 2011). Saving battery power is the most important issue in WSN as each sensor node is energy constraint and it will be dead when it is out of energy. Therefore, to enhance the network life, there are several scheduling techniques to reduce the energy consumption while collecting and sending data to the sensor nodes (Anastasi et al., 2009). All these techniques are not ample for IoT scheduling as IoT Scheduling covers a larger domain as compared to the WSN scheduling techniques.

To learn about the diverse scheduling techniques over this large domain, first we need to understand the resource scheduling process in an IoT environment.

3. Resource scheduling process in IoT environment

Resource scheduling is required at different levels in IoT eco-system, whether it is at data center level, integrated cloud level, any user application level or sensing level. The entire IoT scheduling process goes through a number of necessary steps as depicted in Fig. 3. Data storage in an IoT system can be done on cloud, databases or small data sessions. This data storage mechanism requires scheduling which can be done using the services like broker (in some specific network architecture, a node can be selected as broker through which data transfer and communication occurs between several other nodes in order to avoid multiple links). At the heart of IoT, there are sensors from which data are collected based on which desired actions can be taken. The sensing process also requires scheduling for efficient utilization of the sensing devices. Each of the components in the diagram (such as Broker, High Performance System, User Interface, API, System Interface, and Data Storage System) depicts that this component requires scheduling for improvements in specific functionalities. There are user interfaces through which IoT users can interact with the IoT enabled sensor devices whenever they need some IoT services. Scheduling is required at these interfaces for proper queueing of the incoming requests. The cloud service provider efficiently returns back the query result associated with the database to the user by allocating virtual machines (VMs) and specific data sessions. There are also some APIs through which IoT users can change or modify some decision making processes of IoT services. Another component of IoT service system is system
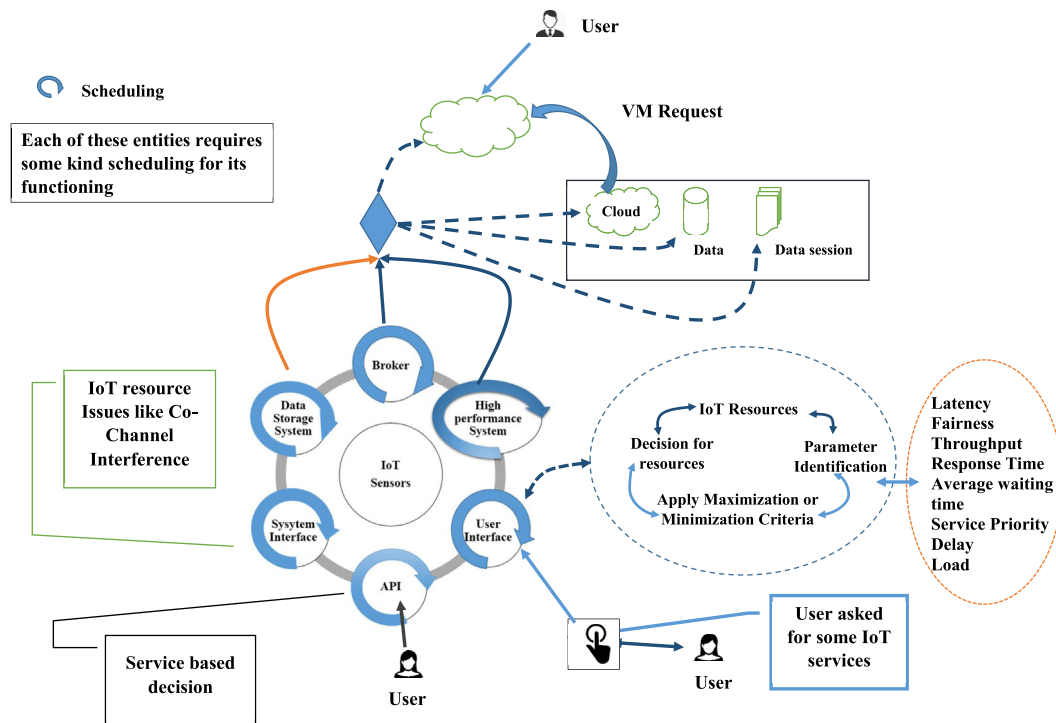
Fig. 3. Scheduling process in an IoT environment.

interface that automatically handles IoT resource issues like channel congestion in IoT network, co-channel interference, delay, (Liu et al., 2014b; Vassaki et al., 2016) etc. For functioning of each of these components, an effective scheduling process which consists of four primary steps such as *resource identification, parameter identification, apply minimization or maximization criteria and decision about the resources*, is mandatory. Therefore, each component involves specific scheduling schemes in order to efficiently handle the associated process and to achieve maximum benefit for the different aspects of IoT resource scheduling. Fig. 3 shows this entire phenomenon of resource scheduling in an IoT environment.

4. Aspects of resource scheduling in IoT

Resource scheduling in IoT depends on several aspects of IoT environment. The service provider selects appropriate scheduling schemes based on the aspect that should be considered for a particular application or user request purpose. Also, the scheme should be so efficient that the accuracy of outcome can reach the optimum level of users' expectations as well as it should provide a good revenue and valuable reputation to the developers and service providers. So far from the research related with IoT scheduling, three clear aspects have been pointed out as shown in Fig. 4; which are *quality of service, IoT architecture, and IoT network infrastructure*.

4.1. Quality of service (QoS) in IoT

A non-functional component, QoS of a system or service provider can be defined as the potential of providing services up to a level of satisfaction. As an emerging dynamic technological paradigm, IoT involves a vast amount of heterogeneous devices, enormous complex data, and actions such as collection, processing and storing. So, IoT needs an optimal QoS architecture to provide excellent service quality. Even though a variety of QoS models have already been evolved for traditional wireless and cloud based networks (Al-Fagih et al., 2013; Li et al., 2012a), but these are not sufficient for dynamic IoT environment. Therefore, there is an immense need to reconsider the essential QoS

attributes such as *Information Accuracy, Energy Consumption, Network Life,* etc. for IoT which is far advanced than traditional QoS attributes, for example, *Response time, Delay, Throughput, Average Waiting Time,* etc. Providing all these QoS to the IoT applications requires specific scheduling strategies which will provide flow isolation and end-to-end guaranteed quality of service (Claudio, 2013). However, a single scheduling scheme is not sufficient due to the features of distinct quality of services of each service. Accordingly, selecting a convenient scheduling algorithm is very critical to provide QoS in order to achieve the desired performance in IoT network.

4.1.1. QoS based scheduling

Task scheduling with the guarantee of QoS is very crucial in real time dynamic IoT environment. This section covers various scheduling techniques which have considered *traffic, load, energy, response time, waiting time, information accuracy, blocking probability* and *throughput* as prominent parameters for QoS optimization. There are mainly four types of resources; *virtual resources, network resources, energy resources and application layer resources* that should be managed efficiently to improve QoS in IoT eco-system. To manage these resources different favorable techniques have been adopted based on which effective scheduling schemes have been proposed. Fig. 5 reflects different types of QoS scheduling techniques.

(a) Virtual resource management: Management of virtual resources involves creation, allocation and removal of the virtual resources. IoT can be integrated with cloud computing environment in which virtual resources like virtual machines exist. A very well-known optimization technique called Particle Swarm Optimization (PSO) can manage the virtual machines while *allocating workload* over a set of available virtual resources. Load Balanced Particle Swarm Optimization (S and P, 2015) scheduling technique dynamically schedules heterogeneous tasks to the relevant virtual machines and also monitors *load* in such a manner so that it can be distributed equally across machines. Hence, in addition, a reduced overall *response time* can be achieved.

Another approach (Moschakis and Karatza, 2015) uses Simulated Annealing in which multiple clouds are available in the IoT eco-system. In the multi-cloud model, clouds are connected with a global dispatcher (GD)
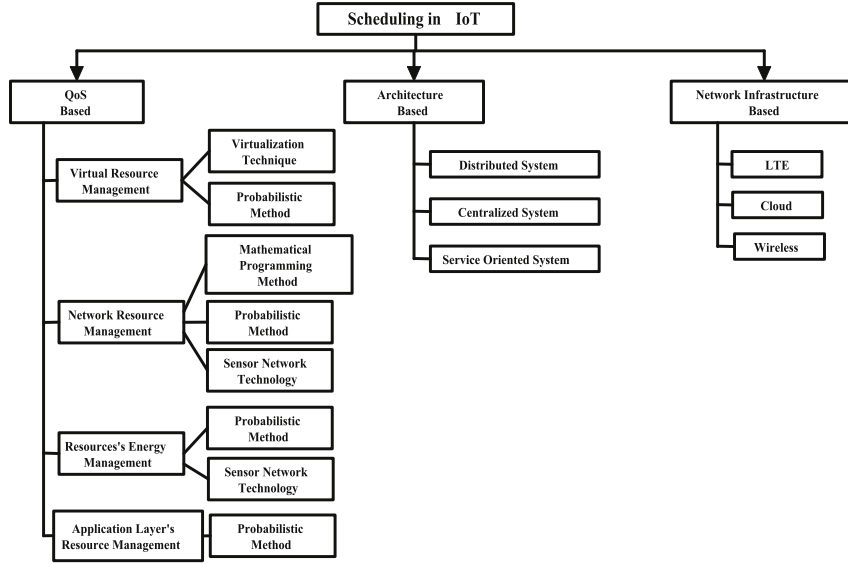
Fig. 4. Primary aspects of IoT scheduling.

and each cloud can have its own cloud dispatcher (CD) through which IoT jobs are distributed among different clouds for computations. The number of available virtual machines (VMs) in each cloud is modeled by random normal distribution so that the dynamic nature of cloud system can be reflected. To model IoT workloads, with few modifications, Lublin99 (Lublin and Feitelson, 2003), a trace generator, is configured to process the vast amount of IoT data in semi-real time order.

There are a few techniques which can provide services for CPU mapping using service level agreements (Singh and Viniotis, 2017). Techniques can also be used to manage virtual resources for multiple tenants in cloud environment. Service level agreement (SLA) which is used to manage the service provided to the user; can be applied with the message arrival pattern information to provide the better resource allocation and decision making. Using these techniques virtual resources can be managed in an IoTCloud integrated environment.

(b) Network resource management: There are a variety of network services available in an IoT eco-system; these include delay sensitive services, real time services, prioritized services, and so on. Network services can be mapped to the network traffic requirement, spectrum utilization requirement, sensing requirement, etc. Scheduling for network resources is required to earn the following objectives:

· Reduce communication cost.
· Efficient management of sensing expenses.
· Maintaining delay sensitive services.
· Managing the real time services and reducing blocking probability.
· Efficient spectrum sharing.

Mathematical programming model can be used to set a trade off between different network parameters in order to achieve the target objectives. The overall throughput at any sensor node can be modeled using mathematical programming method (Li et al., 2014). Let, nodes in sensing layer are denoted by $n_j(j = 1, ..., N)$, and $m_i(i = 1, ..., M)$ is the i_{th} node in network layer, the bandwidth allocation function is denoted as B_{ij} and P_{ij} is the overall power consumption, then the throughput at node n_i can be obtained as:

$$r_i = \sum_{j=1}^{N} (1 - \varepsilon_{ij})\varphi_j\beta_{ij} \log\left(1 + \frac{\hat{g}_{ij}w_{ij}}{\beta_{ij}}\right) \tag{1}$$

where, ε_{ij} is average bit error rate over the link (i, j), the system efficiency is denoted by φ_j, the assigned bandwidth to link (i, j) is β_{ij}, $\hat{g}_{ij}$ is communication channel acquisition, and w_{ij} is transmission power of node m_i. Further mathematical programming model can be used to improve the information accuracy, energy consumption, and some other parameters as shown in Fig. 6.

Real time IoT applications, and delay sensitive IoT data need an instant delivery. Therefore, it is necessary to reduce the *blocking probability* of services. In this context, the QoS alert scheduling model (Sharma et al., 2015) can be used which classifies the incoming traffic into *prioritized* and *non-prioritized traffic*, and stores the first one in a high priority queue, and the second one in a low priority queue with weights assigned to each queue (Geng et al., 2012). A probabilistic Markov chain based Continuous Time Markov Chain (CTMC) model can be used to represent this *weighted prioritized system*.

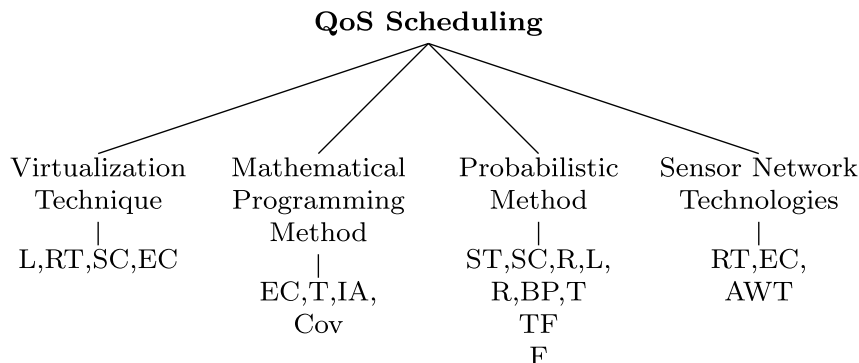Another approach, a service centric continuous time Markov Chain

QoS Scheduling

Fig. 5. Classification of different QoS scheduling techniques.

```
Sensing Layer
   └── Information Accuracy
          └── Data Accuracy
          └── Sensing Time Accuracy
          └── Spatial Accuracy
          └── Reduce data Redundancy
          └── Data Packaging
   └── Energy Consumption
          └── Sleep Management
          └── Life Time Management
   └── Coverage
          └── Sensing Area
```
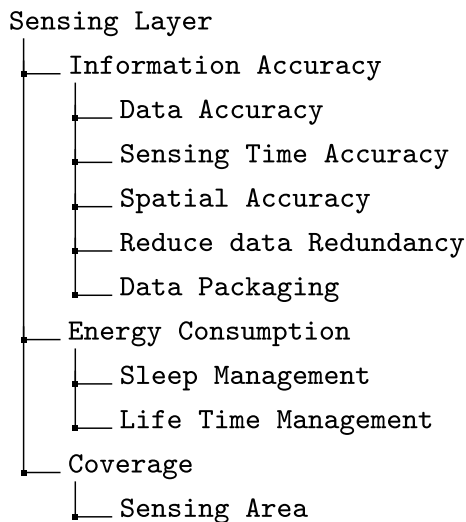
Fig. 6. QoS activities in sensing layer using Mathematical Programming Model.

(CTMC) traffic modeling can be used to calculate the *overall spectrum utilization* and the two important QoS dynamics, i.e. service completion time and queueing delay for both real and non-real time services (Sharma et al., 2015; Eswaran and Bapat, 2015). Further, a probabilistic prediction based joint packet scheduling scheme (Sharma et al., 2015) can be used for dynamically allocating *bandwidth* and reducing the *waiting time* of high priority delay sensitive IoT traffic as well as maintaining a tolerable waiting time limit for other services also. A hybrid scheduling scheme (Li et al., 2012b) uses preemption for delay sensitive traffic, but it is not granted for non-delay sensitive traffic. This is evident that delay sensitive traffics are of high level of importance; therefore it is necessary to grant preemption for these traffics so that the execution can be done as soon as possible. And as the preemption scheme results in a longer length of waiting queue, it is convenient not to grant preemption for non-delay sensitive traffic. Thus, it is possible to reduce the length of waiting queue, and avoid the overflow condition.

(c) Resource's energy management: As sensor nodes are energy constrained devices, so, scheduling process must concern about less energy consumption during message passing. To improve the overall IoT system efficiency, the direct links from sensor nodes to the sink can be minimized as it will reduce the *service response time*, and *energy consumption* in a brokered IoT architecture (Abdullah and Yang, 2013a). In this architecture, each IoT subgroup consists of a set of sensor nodes. Within a subgroup, there is a broker, and every time a new sensor node is selected as a broker. This selection is done with LEACH protocol (Tyagi & Kumar, 2013). It is a self-organizing, adaptive clustering protocol which is used to distribute energy load evenly among sensors within a sensor network with the help of randomization technique. After receiving the messages from a particular IoT subgroup, the broker compresses and amplifies the signal for sending the messages safely to the destination. To model the arrival rate and service rate of the messages from sensors nodes within IoT subgroups, M/M/1 queueing Model (Adan and Resing, 2015; Abdullah and Yang, 2013b) can be used. It represents a system like a queue where there is only one server for processing. Here, jobs arrive like Poisson process and their service times have an exponential distribution. The QoS-aware scheduling scheme, executed at each broker level, is used to calculate traffic intensity using arrival rate and service rate of the incoming messages. The incoming messages are sent to the queues for processing according to the priority. The messages are classified into two classes; emergency messages (EM) those are of high priority messages and best efforts (BE) which are non-critical messages. The primary aim of the scheme is to reduce the message delivery time which in turn reduces the service response time and overall energy consumption.

Furthermore, to retain continuous and non-disruptive flow of services served by the things/sensors, it is necessary to deal with faulty or failed nodes. If the faults or failure can be identified on time then the initial recovery or replacement will be possible which helps to improve overall system efficiency. Using this concept, an extended version of (Abdullah and Yang, 2013a) considers node failure during energy efficient scheduling of messages within IoT subgroups (Abdullah and Yang, 2014). The concept is whenever a node is detected as faulty, first try to repair that node using techniques in (Zhang et al., 2012; Munir and Gordon-Ross, 2011). If, the node cannot be repaired, then a back up node selection procedure finds the right level of back up nodes taking failure rate and already set up IoT subgroups configurations as input. Thus, the method checks the system's overall energy deficiency, repairs faulty nodes or selects right number of back up nodes for constant flow of services.

Energy based scheduling can be done using an energy consumption analysis of the virtual machines, task execution, and available physical resources (Xing et al., 2017). Energy consumption for a complete IoT application can be calculated as a sum of the base energy, task execution energy, idle energy, and virtual machine running energy. Based on these information, scheduling of tasks can be done such that overall energy consumption can be minimized.

While processing soft real time tasks in an IoT environment, the major energy resources are IoT nodes, gateways, and servers. Therefore, two energy efficient scheduling algorithms, i.e. global algorithm and local algorithm (Sarangi et al., 2018) handle the information exchange strategy of different neighboring and distant nodes of IoT network maintaining the required deadlines (loose and tight). Global is preferable when deadline is loose, else, local should be followed. Thus, an optimal energy consumption policy can be achieved.

(d) Application layer resource management: Scheduling the resources in IoT system requires information about the current state of the resources or services which will help further scheduling of these resources. A set of measurement metrics at application layer can be used to get the information about the services and IoT nodes. Based on this information, a QoS manager tries to efficiently schedule the IoT services using a Markov Decision Process (MDP) which converts the overall scheduling problem to a maximization problem (shown in Fig. 7). Fundamentally, MDP (Marini and Walczak, 2015; Givan and Parr, 2001) is a non-deterministic search problem where action outcomes depend only on the current state. An MDP is defined by five-tuple $(Q, A, P, \hat{R}, \delta)$, where, Q is a set of states, A is a set of actions, P is the probability that $a \in A$ from $q \in Q$ leads to q', i.e. $P(q'|q, a)$, $\hat{R}$ is the reward function (q, a, q'), i.e. the immediate reward after the transition
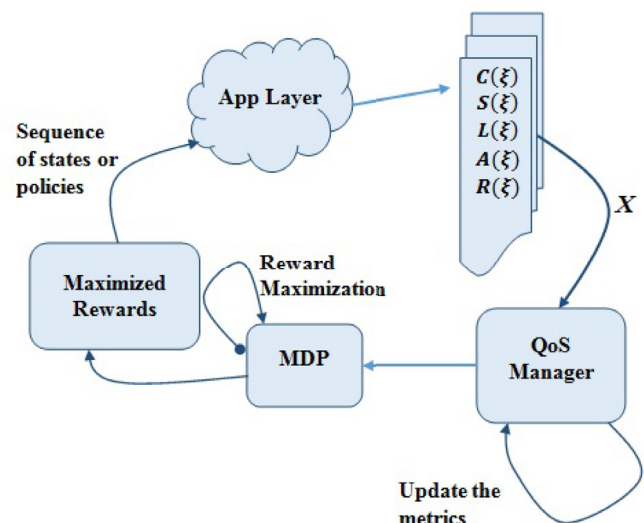

Fig. 7. QoS activities in application layer using MDP model.

from state q to q' and δ is a discount factor $\delta \in [0, 1]$, which is responsible for preferring rewards now than later. The values of rewards decay exponentially and thus a reward which is n steps away is discounted by δ^n. This model is used to find an optimal policy $\pi^*: Q \rightarrow A$ which maximizes expected utility.

Optimal policy can be obtained using equation (2) and equation (3):

$$\pi^*(q) = argmax_{a \in A}\left(\sum_{q' \in Q} P(q, a, q')V^\pi(q') \right) \qquad (2)$$

$$\hat{R}(q, a) = \sum_{q' \in Q} \hat{R}(q, a) * P(q, a, q') \qquad (3)$$

Further optimized policy can be obtained as:

$$V(Q) = \max_{a \in A}\left(\hat{R}(q, a) + \delta \sum_{q' \in Q} P(q, a, q')V^\pi(q') \right) \qquad (4)$$

Using this, a state model, a model of bandwidth change, and an energy consumption model have been proposed in (Li et al., 2014) which is used in QoS-aware services iteration algorithm that aims to find an optimal decision policy for each QoS metric. In the three layer QoS architecture, the considered QoS metrics associated with application layer are service cost, service time, load, reliability, and reputation. These are the most important factors for a service delivery system. Reputation, a new type of QoS metric other than traditional QoS metric, is the trust level of services at IoT nodes. The MDP model is basically used here for decision making in application layer.

Table 2 shows the contrast among several QoS scheduling techniques in IoT environment.

4.2. Architecture based

As far as the literature has been studied, it is evident that there is no fixed architecture for IoT. Therefore, its architecture varies from application to application depending upon the specific needs and other design constraints. Hence, it is required for IoT to maintain an open architecture in order to escalate the interoperability among several heterogeneous systems and distributed resources (Vermesan et al., 2011).

On the basis of available study, we have classified the scheduling techniques depending upon the particular IoT architecture into three major categories; *distributed, centralized, and service oriented* as shown in Fig. 8.

4.2.1. Distributed architecture

In typical IoT applications, there is a lack of proper deployment of distributed IoT devices. Hence, *co-channel interference* makes a severe impact on the overall performance of the network, which in turn results in *coexistence* issues within the network when the IoT nodes are deployed densely. Several techniques have been discovered so far for solving *coexistence* issues. One useful scheme based on listening before talk strategy is Carrier Sense Multiple Access (CSMA) which nowadays is lagging to handle collision probe large IoT network, proper channel utilization, and energy consumption. Scheduling based message passing algorithm can be used for handling the co-channel existence issues in wireless devices. *Message Passing Algorithm for coexistence of IoT Wireless Devices* is a distributed scheduling scheme (Sohn and Lee, 2015) which efficiently deals with the issues mentioned above. It results in improved sum rate and less energy consumption without any centralized scheduler. *Sum rate* is a function of power vector, which requires maximization for efficient performance. It can be calculated using signal-to-interference-noise-ratio (SNIR) as: $R(P) = \sum_{n=1}^{N} \log(1 + SINR_n)$: where, P is a power vector for N IoT device pairs, $P_n = \{0, P\}$ and $SINR_n$ is the signal-to-interference-noise-ratio for nth pair of IoT devices.

Another important issue, i.e. excessive *energy consumption* can be diminished using a decentralized scheduling scheme (Domingo-Prieto et al., 2016) for 6TiSCH networks. The scheme uses a policy based on Proportional, Integral and Derivative (PID) algorithm to dynamically control the *traffic* in the network and reduces the unnecessary *energy consumption*.

4.2.2. Centralized architecture

In the recent era of IoT, there is a need of centralized control in order to manage the large-scale IoT networks, establish complex and secure connections through routers, and simplify user operations in a non-IT environment. It can be done using a Software Defined Network (SDN) that provides centralized network control with dynamic, flexible, and automatic reconfiguration capabilities.

A 6TiSCH centralized scheduling method (Thubert et al., 2015) is a combination of deterministic networking (it is the networking model in which amount of resource consumption can be modeled using the historical information about resource consumption) with SDN. Some typical network related problems in IoT, such as *packet loss* due to *congestion*, *uncontrolled jitter*, and *poor bandwidth* utilization can be avoided if some *pre-computation* and *pre-allocation* of *pre-determined* physical resources can be done; hence deterministic networking model should be used for this purpose. To minimize the *energy consumption* and balance the *load*, architecture 6TiSCH can be incorporated to monitor and manage Time Slotted Channel Hopping (TSCH) network by PCE controller.

4.2.3. Service Oriented Architecture

Development of a Service Oriented Architecture (SOA) is a challenge (Atzori et al., 2010; Li et al., 2015) in case of IoT eco-system; it typically consists of five characteristics (MacLennan and Van Belle, 2014): *discoverable, reuse, efficiency, loose coupling,* and *division of responsibility* which can be used to make the IoT network more flexible and scalable to adopt the future-proof devices. The adaptation of SOA technology enables decomposition of complex and bigger IoT system into simple, well defined and narrow IoT applications. It can act as a middleware between service provider and requester to communicate deliberately in spite of the heterogeneous nature of IoT system, business artifacts and other intricate documents (Bandyopadhyay and Sen, 2011). A broker based technique can be used for managing the user requests and available services in an SOA based IoT system.

Message Stability by Shortest Processing Time Scheduling consists of clients and several IoT units where each IoT unit comprises an aggregator (Leu et al., 2014), i.e. broker which collects all information from different sensor nodes and issues necessary commands to the actuators. Table 3 presents a comparison between various types of scheduling based on different IoT architecture.

4.3. Network infrastructure based

In an IoT environment, it is very important to use the supporting network infrastructure intelligently so that things will be able to manage their transportation individually, full automation of services take place, and optimization of logistics can be done. To realize this perception, the architecture of IoT should be built on the top of network infrastructure which assimilates both the wired and wireless technologies in a translucent and coherent manner. Thereby, WSN has attained more importance due to its wire-free communication with low power and low cost objectives. Recently, LTE and cloud networks are becoming very prominent for the promise to serve scalable, reliable, delay tolerant, and robust services (Vermesan et al., 2011).

4.3.1. WSN

The most important factor in WSN is *energy* as for a single node failure the whole network can be disconnected and all the execution within the network will be aborted. So, management of the available energy in WSN is an important task; efficient scheduling technique like

Table 2
A comparison of QoS scheduling schemes in IoT environment.

| SL. No	Problem	Objectives	Parameter & Metrics	Model	Simulation Tool	Remarks
1	Manage devices' heterogeneity and workload while ensuring QoS requirements	To reduce application response time and balance the load	RT,L	Load Balanced Particle Swarm Optimization scheduling	…	A QoS requirements based multiple level QoS can be considered (S and P, 2015)
2	Optimize overall scheduling performance and minimize resource cost of IoT network	An enhanced QoS architecture according to the requirements at each of three layers	L,EC, RT,AWT, T, IA	MDP and Mathematical Programming Method	Matlab 2009R	Test the proposed method in real time environment (Leu et al., 2014)
3	Message scheduling for QoS requirements	To differentiate emergency messages from non-mission critical messages	AWT,EC	Energy Consumption Model with M/M/1 queueing Model	Matlab version 7.8	Can consider more than two classes of messages for better scheduling (Abdullah and Yang, 2013b)
4	Improve application level message scheduling technique and explore service provisioning in IoT environment	To improve IoT system efficiency	RT,EC	Energy consumption model based on LEACH and TDMA schedule	Matlab version 7.8	A multi-hop routing scheme should be proposed that will also consider the priority of the messages (Abdullah and Yang, 2013a)
5	Investigate the application of simulated annealing in a multi-cloud environment	To achieve significant performance gain and effective cost model	SC	A Simulated Annealing model	Trace generator,DES	Implementation for heterogeneous multi-cloud environment can be done (Moschakis and Karatza, 2015)
6	Optimize the performance of real-time delay sensitive IoT traffic	To reduce blocking probability of prioritized IoT traffic	BP	Probabilistic Markov Chain based CTMC Model	Matlab R2013b	Blocking probability should be reduced for both type of services (Sharma and Kumar, 2015)
7	A service based centralized scheduling algorithm to model IoT traffic	To improve spectrum utilization and reduce overall queueing delay	ST,F	Service Centric Markov Chain Based Spectrum Sharing Method	Esper 4.11.0 (CEP engine)	Effective cost model, channel aggregation and preemption queue analysis can be done (Eswaran and Bapat, 2015)
8	A dynamic allocation of bandwidth to delay sensitive IoT traffic	To reduce the waiting time of different prioritized services	AWT, T	Joint Packet Scheduling and Bandwidth Allocation Model	Matlab R2013b	Optimize the performance to serve better for all types of classes and for average buffer types also (Sharma et al., 2015)
9	A hybrid strategy to analyse queueing delay for prioritized delay sensitive services	To reduce the waiting time for both types of services (delay and non-delay sensitive)	AWT	Hybrid Scheduling Strategy	NS-2	Grant pre-emption for non-delay sensitive services (large in number)(Li et al., 2012b)
10	Resource allocation to IoT application in a cloud environment	To achieve service level agreement	L	Limiting traffic and CPU scheduling	DES	Introduce a broker service (Singh and Viniotis, 2017)
11	Energy efficient resource allocation in IoT-cloud integrated environment	Virtual machine allocation in an energy efficient and fair way	EC, F	By modeling, the physical machine's fairness and IoT application's energy consumption	CloudSim	Consider other parameters like load and priorities (Xing et al., 2017)
12	Optimize energy consumption at IoT nodes, servers, and gateways	To reduce energy consumption as well as meet the deadlines	EC	Global and local energy model with DVFS (dynamic voltage-frequency scaling)	NS3 and CloudSim	Priority based hard real time tasks need to be tackled (Sarangi et al., 2018)

49
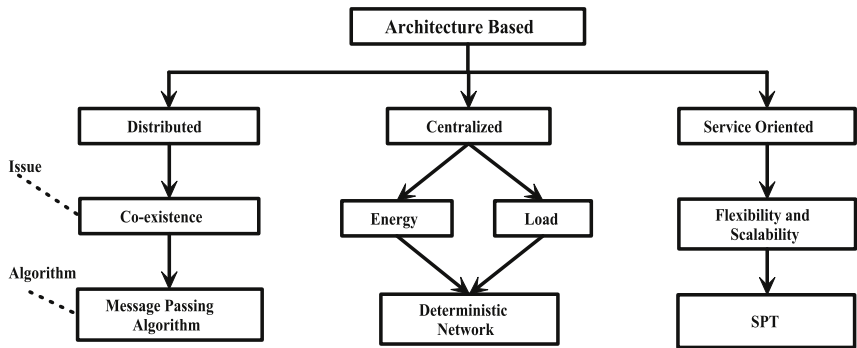
Fig. 8. Scheduling based on different IoT architecture.

TRAP (Task Requirement Aware Pre-processing and scheduling) (Bharti and Pattanaik, 2016), can be used for managing the energy of each node of the network. Query pre-processing can result in better energy management. Each query may be composed of several tasks and sometimes it may happen that two or more queries have one or more tasks similar in functionality and QoS requirements. Therefore, it is important to decompose the queries and group the same functioned tasks in order to reduce the traffic, task completion time and save battery life of the sensor nodes which can result in a long life of the overall network.

4.3.2. LTE

In the recent age of Internet, Long Term Evolution (LTE) networks are becoming the most prevalent connectivity technology in cellular IoT due to its all IP architecture. The cellular IoT requirements include enhanced battery life, low deployment cost, wide coverage and support for an enormous number of devices. Hence, involving LTE network with IoT results in prolong battery life through power saving mode, little device and deployment cost using less complex devices and tolerant shared carrier capacity, extended coverage through advanced coding and so on (NOKIA, 2016; Zarrinkoub, 2014). LTE coverage has already outstripped the 3G coverage; thus LTE is almost 3 times more efficient than 3G and 20 times more efficient 2G. Further, LTE yields built-in security along with the potential to manage the robust and scalable network traffic (Kevin, 2016). Previously, LTE was designed for H2H (human to human) communication but nowadays, LTE, which is a 4G technology, can meet the challenges of M2M communication and provide a cost effective solution by the convergence with WSN (Crosby and Vafa, 2013). M2M generates huge amount of traffic which results in channel congestion and complexity in this network. These issues can be taken care by using scheduling techniques, so that overall performance of the system can be increased.

Fixed Access Grant Time Interval (AGTI) is a time controlled scheduling scheme which schedules M2M traffic in LTE network and reduces *channel congestion* and *complexity*, caused by Machine Type Communication Devices (MTCDs). But it follows a strict resource allocation policy that results in inefficient utilization of resources. To overcome this issue, *dynamic AGTI* time controlled scheduling scheme (Edemacu and Bulega, 2014) has been developed which provides optimal QoS requirements for both H2H and M2M traffic in IoT scenario by dynamically allocating the sub-frames of time frequency resource blocks.

Power consumption and *radio spectrum management* are also two critical issues in an IoT network. A fuzzy-based power saving scheduling technique (Kuo and Chou, 2015) for an LTE/LTE-Advanced enabled IoT network can efficiently deal with these two issues. This scheme also considers the overall performance of the entire system by limiting unexpected delay for IoT over LTE networks through exact DRX (Discontinuous Reception (DRX) is used to provide techniques for user equipments (UEs) to supply micro-sleep mode even in the active state in order to save power consumption in LTE network. It also ensures QoS and end to end connectivity) cycle timer. To ensure the real time requirement of IoT, a guaranteed scheduling latency which consists of the context switching time and scheduling decision time, can be used.

To maximize network *throughput* and provide dual connectivity among IoT devices in a two-tier heterogeneous wireless cellular IoT network like LTE, three effective scheduling schemes (Kim, 2016) efficiently handle three important issues such as *load balancing, radio resource management* and reduction of *co-channel interference* among small cells in the network.

4.3.3. Cloud

Resource virtualization is the key factor in cloud computing. Scheduling of these cloud resources is required when it is integrated with the IoT devices (Kim et al., 2016). Virtual machines are the main resources in a cloud environment and need to be managed efficiently. Requests, generated from the users, are considered to be of different classes and require different types of handling mechanism.

A heterogeneous Dynamic Dedicated Server scheduling algorithm (Narman et al., 2016) can effectively improve the performance in terms of *throughput, drop rate* and *resource utilization* of both the shared and dedicated server system taking into account the *priority* of the incoming IoT requests in a cloud environment. In this algorithm, each request class has been assigned different groups of dedicated servers with a

Table 3

Comparison between different IoT architecture based scheduling schemes.

SL. No.	Objective	Architecture	Parameter & Metrics	Simulation Tool	Remarks
1	To maximize sum rate	Distributed	T	Matlab	A centralized scheduling scheme need to be introduced for better performance (Sohn and Lee, 2015)
2	To control network traffic	Distributed	EC	OpenWSN (Watteyne et al., 2012)	Analysis of the blocking probability in case of prioritized requests can be done (Domingo-Prieto et al., 2016)
3	To gain global optimization for low power WSNs	Centralized	L,EC	Practical Network topology with PCE	Technique for heterogeneous SDN to obtain more gain should be proposed (Thubert et al., 2015)
4	To maintain stability in response messages	SOA	RT	RT-OSGi framework	Need to implement the scheme for all types of web browsers (Leu et al., 2014)

number of required virtual machines (VMs) and this assignment process regularly updates the number of assigned servers for each class. To model the entire scenario, the arrival rate of incoming requests follows Poisson distribution and service time of requests maintains exponential distribution. Based on the results, it has been observed that there is less influence of priority levels on performance for both types of system when traffic flow is low, when traffic flow is high, priority level affects the throughput significantly for homogeneous servers, and substantially, by assigning appropriate priority levels, heterogeneous system can perform better than homogeneous system. Whereas, a sudden change in IoT sensing environment can generate immediate resource need for IoT applications running in cloud platform. So, to avoid service degradation, a resource aware VM migration approach (Paulraj et al., 2018) is used to meet the sudden resource requirements of IoT applications. While keeping the incoming data from IoT applications intact, this approach provides minimum migration time and downtime than state of the art approaches.

Various network based IoT scheduling techniques referred in this review have been shown in Fig. 9 and Table 4 presents the comparison among them.

5. Analysis

IoT resource scheduling covers a broader area of scheduling techniques while considering different metrics like load, response time, service time, waiting time, throughput, energy efficiency, information accuracy, service cost, performance, and so on. Based on the available techniques, discussed in the previous section, a detailed threefold analysis has been done in this section. The first subsection 5.1 presents a metrics-based analysis of QoS. A comparison among several scheduling techniques based on different queueing models and empirical study has been shown in the next subsection 5.2. The last subsection 5.3 edifies the simulators available to validate the scheduling techniques.

5.1. Metrics based analysis

5.1.1. Load

Load is the measurement of usage of IoT service which can be represented as $\lambda\upsilon$, where λ is request arrival rate and υ is service rate (Li et al., 2014). The Load Balanced Particle Swarm Optimization (S and P, 2015) technique iteratively assigns the workload among virtual machines in order to maintain a balanced task execution condition. Whenever a new service request arrives, to balance the load among virtual machines, the currently executing tasks are rescheduled. Thus, it avoids the situation where some resources are heavily loaded and some are underutilized. Load can also be managed using some deterministic modules (Thubert et al., 2015) in an IoT network which can help in scheduling of the IoT devices.

5.1.2. Response time

Response time is one of the basic QoS measures for any type of services. Using the message scheduling algorithm (Abdullah and Yang, 2013a) with shortest processing time first (SPT) rule, the service requests in IoT environment can be served which efficiently balance the requests in each subgroup. It can provide stability of responses which also results in less consumption of energy. Response time can also be reduced significantly using swarm optimization technique (S and P, 2015) by scheduling heterogeneous tasks in appropriate processing machines. Broker and queues (Leu et al., 2014) can be used for decreasing the response time of the system, and further scheduling of queues can be done for better processing of the computing systems.

5.1.3. Service time

It is the time duration from accepting a request to successfully completion of the task. One way to use this as a metric is to implement it using a queue. If there are k messages which are batch processed by IoT nodes as a $M/M/k$ queue model, then the expected service time can be defined as (Li et al., 2014) $E(S(t)) = (\rho\Pi_k)/((1 - \rho)^2\lambda + 1/\varepsilon)$, where, $\rho = \lambda/(k\varepsilon)$ is server utilization, λ is request arrival rate, ε is service rate. The CTMC-IoT algorithm (Eswaran and Bapat, 2015) can improve job servicing rate by 50% over traditional algorithms like Own Criticality Based Priority (OCBP) and 18% over previously proposed SPOC (Eswaran et al., 2014) algorithm.

5.1.4. Fairness measure

Fairness in terms of network engineering is the fair sharing of system resources among the service requests in order to process all types of requests giving equal importance. The CTMC method (Eswaran and Bapat, 2015) achieves almost same completion time for 300 services of 4 different types that are uniformly distributed. Thus, it shows improved result in spectrum utilization which is the total number of channels used by all the different types of services, than another fairness algorithm, Max-Min fairness (Nace and Pióro, 2008). A virtual machine scheduling (Xing et al., 2017) method efficiently schedules the IoT applications in cloud environment maintaining the overall resource fairness using VM migration strategy.

5.1.5. Waiting time

Before processing any service request, it is stored in the waiting queue. If the number of requests is too large, the length of waiting queue is extended and as in general each waiting queue has a fixed size, overflow may occur. As a result, packet loss may happen frequently. Therefore, it is important that the length of waiting queue should be reduced. The IoT traffics can be delay sensitive or non-delay sensitive. Delay sensitive traffics are of utmost importance, therefore, their waiting time must be less and hence preemption is allowed for these
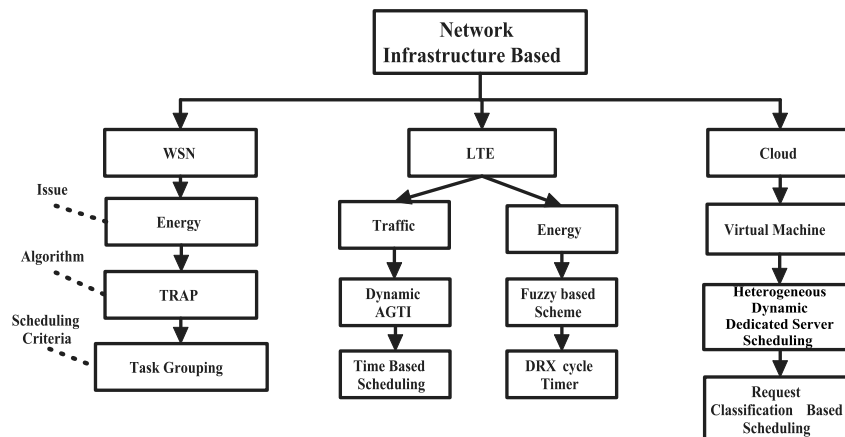

Fig. 9. Scheduling based on different network infrastructure.

Table 4
Comparison between network based scheduling techniques in IoT.

Sl. No.	Objective	Network	Parameter & Metrics	Simulation Tool	Remarks
1	To reduce network traffic and energy consumption	WSN	D,EC, AWT, L, SC	OPNET 18.0 modeler	Need to handle the tasks that have been executed partly and involve other QoS requirements (Bharti and Pattanaik, 2016)
2	To manage network traffic with efficient resource utilization and reduce blocking probability	LTE	Network traffic, BP	LTE System level simulator	Analyze the result for multiple M2M and H2H traffic and prepare a real time scenario to test the effectiveness of the method (Edemacu and Bulega, 2014)
3	To enhance the overall IoT network performance by reducing power consumption and effective managing of radio resources	LTE	EC,T,LT	Matlab	Manage the trade-off by adding more parameters (Kuo and Chou, 2015)
4	To provide Dual connectivity in wireless cellular IoT network	LTE	L,CCI,T	LTE System level simulator	Examine the stability of the schemes with respect to mobility and varying user traffic demand. And analyse the performance and complexity with respect to time and money (Kim, 2016)
5	To improve and analyze the performance of cloud systems by considering homogeneous and heterogeneous servers, and priority classes of IoT requests	Cloud	Drop rate, T, RU	Matlab	Need to investigate the effects of heterogeneity levels of servers and IoT requests on cloud computing performance using Gini Index (Narman et al., 2016)
6	To achieve minimum migration time, downtime, and delay	Cloud	D	Real time Hadoop framework	Enhance the energy efficiency of cloud data centre through decision making optimization technique (Paulraj et al., 2018)

traffic. But through the analysis method (Li et al., 2012b), it has been observed that for preemption scheme, the length of waiting queue is increased gradually which is not desired due to packet loss. Therefore, it is better to keep preemption scheme for only delay sensitive traffics and rest can be treated in the normal way. On the other hand, the simulation results of a probabilistic prediction based approach (Sharma et al., 2015) show that the waiting time for high priority traffic with buffer size 4,6 and 10 is same, for medium priority traffic with buffer size 20,50 and 100, it is almost same but more than high priority traffic and when buffer size is greater than 100, i.e. for low priority traffic, the waiting time is more than previous two. With the increment of variable sized packets, waiting time is also increased as it will be scheduled only when required *bandwidth greater than or equal to allocated bandwidth*. So, it is clear that buffer size has a direct proportional impact on the average waiting time. Better result can be achieved by keeping tolerable limits for waiting time. In QoS message scheduling scheme (Abdullah and Yang, 2013b), when there is a large number of IoT subgroups, the waiting time for the messages is increased and it is reduced when number of subgroups become less. For best effort messages, when there are 10 subgroups, delay is 50 μs and for 5 subgroups, delay is 10 μs. For high priority messages if number of subgroups are 10, delay is 10 μs and for 5 subgroups it is almost 0, i.e. no delay. Therefore, more subgroups result in more waiting in the waiting queue.

5.1.6. Throughput

Maximization of throughput results in better service; buffers can be used to increase throughput in an IoT system. According to the probabilistic prediction based method (Sharma et al., 2015), the throughput remains high for all the three types of prioritized services when buffer size is greater than 10, 100 and 100 respectively. It has been observed that the throughput is decreased for medium and low priority traffic when buffer size is less than 4, 20 and 20. But, as the high priority traffic always gets the advantage, so throughput remains consistently high for this. Therefore, to get high throughput it is required to keep the buffer size bigger, and hence to achieve optimal performance it is important to keep the buffer size within a certain limit so that average waiting time can be reduced as well as throughput is increased. A distributed scheduling method (Sohn and Lee, 2015) to increase the overall sum rate or throughput for coexistence of IoT wireless devices shows better performance than no-coexistence and CSMA approach. Another approach (Kim, 2016), to maximize the throughput of IoT cellular network, a packet forwarding scheduling algorithm uses backpressure (Kim et al., 2010) algorithm and outperforms other different packet forwarding methods such as Round Robin. It achieves 20 percent better performance than others. Considering the heterogeneity and priority levels of IoT application requests, a cloud based scheduling approach (Narman et al., 2016) shows that overall throughput can be increased significantly upto 40% than DSS (Delimitrou and Kozyrakis, 2013) method under heavy traffic flow if the processing servers are of heterogeneous types instead of homogeneous servers.

5.1.7. Energy consumption

Energy is a scarce resource in IoT environment. It is one of the most dominant and challenging factors for any IoT application development. An effective an energy consumption model (Abdullah and Yang, 2013b; a) presumes that all the sensor nodes have same energy at the starting phase and gradually get reduced as long as they are executing. Here, the notion of service differentiation is imported in order to monitor the energy conservations among IoT nodes. In the simulation, for BE messages when 10 IoT subgroups have been created, the node to node delay is almost 50 μs and if there are 5 IoT subgroups then delay is 10 μs. Where for EM messages delay is 10 μs for 10 subgroups and almost 0 for 5 subgroups. So, it is clear that we can lower the delay if number of subgroups becomes less. On the other hand, if the number of brokers is very small, a huge loss in energy is incurred because of two important reasons. First, when a small number of brokers compress a massive amount of data, generated by a
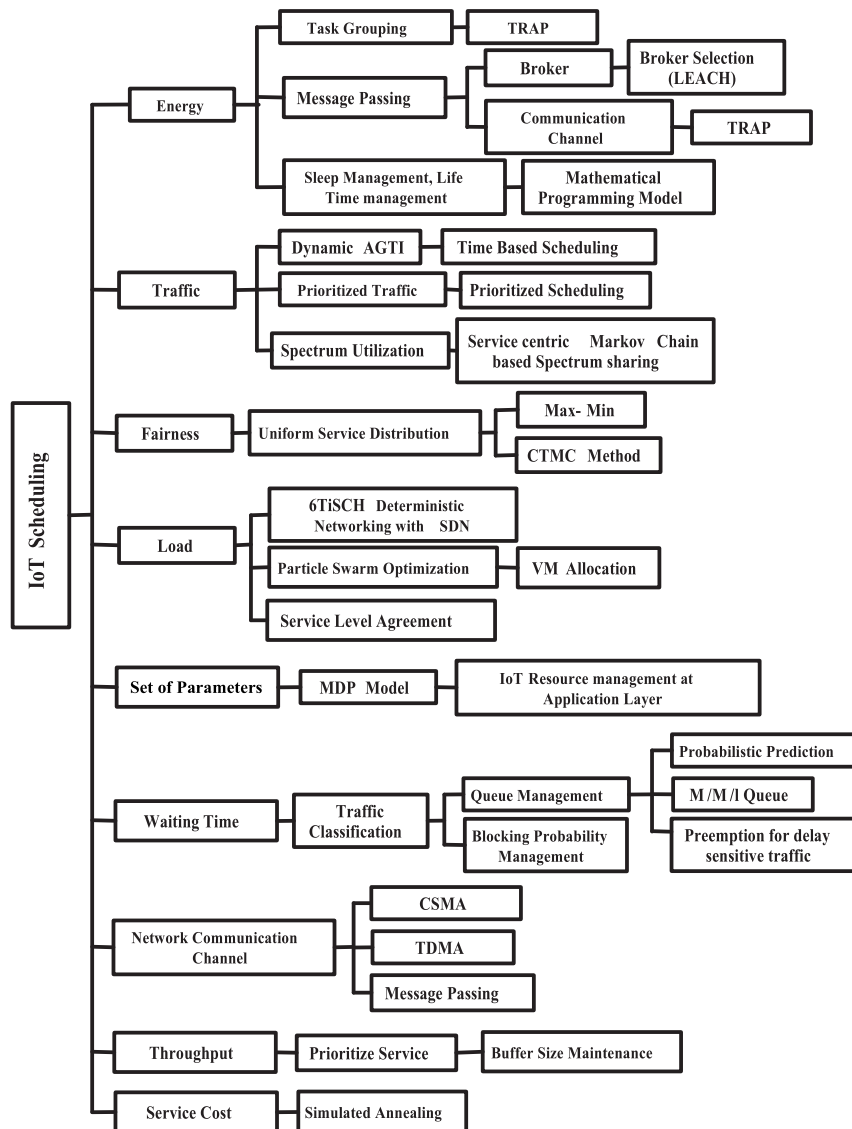
Fig. 10. Parametric analysis of IoT scheduling.

large number of nodes, a significant amount of energy consumption is taken place. Second, as there are very few brokers, most of the nodes remain at a far distance from the respective broker and hence consume more energy for data sending. In opposite to this, if there are a large number of brokers; the overall energy consumption will also increase. So, it is very convenient to set a proper percentage of brokers. In the first round of simulation total enduring energy by BE messages is 6000 J and in the subsequent simulation phases, it is reduced up to 20 J to 30 J. Where, for EM messages it is 350 J in the starting phase and going down towards almost 90 J. According to the proposed QoS decision scheme (Li et al., 2014), when there are 100 nodes in an IoT network and 10 service requests are carried out, the first dead node occurred after 2334 s. And if there is no QoS scheduler, first dead node occurred after 415 s. Therefore, it is very evident that using this scheme, the energy consumption gets reduced. As a result longevity of battery life gets increased and this will result in a long life of IoT network. Channel congestion and co-channel interference can also result in high energy consumption; handling these issues (Sohn and Lee, 2015) with proper scheduling of channels in co-existing networks can reduce energy consumption. Classification based on service requirements (Bharti and Pattanaik, 2016) and then scheduling the services can also result in better energy consumption. For an LTE enabled IoT network, a fuzzy-based power saving scheduling scheme (Kuo and

Chou, 2015) can reduce energy consumption almost by half using DRX cycle timer. While scheduling IoT applications in cloud environment, a virtual machine scheduling approach (Xing et al., 2017) can work well to cut down the energy consumption and the energy consumption analysis shows the better result than the benchmarks. On the other hand, in order to reduce the energy consumption in 6TiSCH network, a distributed PID based scheduling scheme (Domingo-Prieto et al., 2016) decreases the unwanted traffic load without using a centralized management entity.

5.1.8. Information accuracy

To reduce data redundancy and transmission load of dynamic real-time application, information accuracy is one of the most important QoS metrics in sensing layer. It can be significantly improved using the method in (Li et al., 2014). It has been shown that if there are less than 13 IoT nodes and 10–20 services for processing, the error rate between the captured data and the actual real data is 5% when number of services is up to 17. The error rate increases rapidly if the number of services is more than 17. Thus, according to the proposed method, for a fixed number of processing nodes, the accuracy rate is improved when a certain number of requests are served and it decreases if the number of requests is more than some threshold.

5.1.9. Service cost

With the simulated annealing approach (Moschakis and Karatza, 2015) a significant performance gain has been observed throughout the experimental analysis. Along with the performance, an effective cost model has also been achieved. Several comparisons have been done under different situations between Shortest Queue First (SQF) and the annealing approach technique and it has been found that the overall cost is always less than the SQF.

5.1.10. Blocking probability

In order to reduce the packet loss, the blocking probability should be decreased. Complete traffic in IoT network can be prioritized for increasing throughput, better channel utilization and congestion reduction. Using prioritized (Sharma et al., 2015) traffic, the emergency traffic is given high priority, and for a limited bandwidth there can be a significant reduction in the blocking probability of these types of traffics in case of congested network also. On the other hand, blocking probability will be increased for non-priority traffic in order to maintain the balance in bandwidth allocation. The technique can also be implemented using a time based access to the communication channel (Edemacu and Bulega, 2014). Thus, when there is continuous increment in load for both prioritized and non-prioritized traffic, this method efficiently reduces the packet loss.

A parametric analysis of IoT scheduling techniques has been shown in Fig. 10. The tree like structure depicts the metrics of overall IoT scheduling process and the necessary methods for solving purpose. Frequency of the parameters used for scheduling approaches can be understood using the Fig. 11. Table 5 indicates the leading metrics used in three primary aspects of IoT scheduling.

5.2. Comparison among several resource scheduling techniques

The overall problem of scheduling can be considered as a maximization of resource efficiency, where the performance of the entire system needs to be maximized. Different types of modeling techniques have been used to model different resource management scenarios. Memoryless property of the Markov chain model can be used to model requests with the mutually independent service time and with independent inter-arrival time. Markov chain consists of multiple priority queues in which queue selection is important based on a selection criteria as described in Table 6. When there exists one single server, then $M/M/1$ queue model can be used to model the scenario. Here, resources are allocated to the queued service requests based on queue priority. If there are heterogeneous resources available for scheduling then, for an equal number of requests and servers, $M/M/m/m$ model can be used, while $M/M/m/N$ model can be used in case of m number of servers and N number of requests in the system. Using $M/M/m/m$ and $M/M/m/N$,

Table 5

Notable metrics of three primary aspects of IoT scheduling.

Classification	References
QoS	
Virtual resources	
Load	(S & P, 2015; Singh and Viniotis, 2017)
Energy Consumption	(Xing et al., 2017)
Response Time	(S & P, 2015)
Network resources	
Delay	(Eswaran and Bapat, 2015; Sharma et al., 2015; Li et al., 2012b)
Blocking Probability	(Sharma et al., 2015)
Throughput	(Li et al., 2014)
Information Accuracy	(Li et al., 2014)
Energy Consumption	(Li et al., 2014)
Average waiting Time	(Li et al., 2012b, 2014; Sharma et al., 2015)
Energy resources	
Energy Consumption	(Abdullah and Yang, 2013a, 2014; Xing et al., 2017)
Average Waiting Time	(Abdullah and Yang, 2013b)
Response Time	(Abdullah and Yang, 2013a)
Application Layer resources	
Service Cost	(Li et al., 2014)
Service Time	(Li et al., 2014)
Load	(Li et al., 2014)
Architecture Based	
Distributed	
Co-Channel Interference	(Sohn and Lee, 2015)
Throughput	(Sohn & Lee, 2015)
Energy Consumption	(Domingo-Prieto et al., 2016)
Service oriented	
Response Time	(Leu et al., 2014)
Centralized	
Energy Consumption	(Thubert et al., 2015)
Load	(Thubert et al., 2015)
Network Infrastructure Based	
WSN	
Energy Consumption	(Bharti and Pattanaik, 2016)
Delay	(Bharti and Pattanaik, 2016)
Load	(Bharti and Pattanaik, 2016)
Average Waiting Time	(Bharti and Pattanaik, 2016)
Service Cost	(Bharti and Pattanaik, 2016)
LTE	
Blocking Probability	(Edemacu and Bulega, 2014)
Energy Consumption	(Kuo and Chou, 2015)
Throughput	(Kuo and Chou, 2015; Kim, 2016)
Latency	(Kuo and Chou, 2015)
Load	(Kim, 2016)
Co-Channel Interference	(Kim, 2016)
Cloud	
Load	(S & P, 2015; Narman et al., 2016)
Throughput	(Narman et al., 2016)

modeling of the heterogeneous resources and services can be done. To model a general service time distribution, M/G/m/m model can be used. Optimization techniques result in the best possible solution from the available solutions of resource allocation. Techniques like, Particle Swarm Optimization (PSO), Markov Decision Process (MDP), Simulated Annealing can be used, where the overall objective of effective resource allocation is achieved. The comparison of several existing scheduling techniques based on their modeling strategies and decision making mechanism is shown in Table 6.

5.3. Simulation model analysis

Simulation of an IoT scheduling requires to have a job or service arriving model, waiting queues, service selection model, parameters to evaluate the scheme, and objectives that need to be achieved. Well established models like Markov chain, Queueing system, SDN based model, or some empirical model can be used for these purposes.

For communication between IoT devices, which can be one to one or many to one, message passing algorithms can be used. Further, these approaches can be simulated using subgroups of nodes and maintaining

Fig. 11. Prominent parameters in IoT resource scheduling.

Table 6
Modeling strategy and working principle based comparison of reviewed scheduling techniques.

Model	References	Scheduling Criteria	Working Principle and Working Mechanism
Markov Chain	(Sharma and Kumar, 2015)	Bandwidth prediction	Queue Selection is done based on the average queue length · Service classification is done based on priority (High or Low) · At application and network layer
Markov Chain	(Eswaran and Bapat, 2015)	Priority Based	Queue Selection is done based on priority index (depending upon service priority and tolerable delay) · Service classification is done based on dynamic priority · At application and network layer
Markov Chain	(Sharma et al., 2015)	Bandwidth Prediction	The weight assigned to each queue is updated when average queue length increases more than a threshold value · Service classification is done based on priority (High or Medium or Low) · At application and network layer
M/M/1	(Abdullah and Yang, 2013a)	SPT	Service rate allocation is done based on the traffic intensity; if traffic intensity is greater than a threshold then SPT is applied · Service classification is done based on the types of the messages · At broker layer
M/M/1	(Abdullah and Yang, 2014)	SPT	Service rate allocation is done based on the traffic intensity and also optimally selects the number of nodes for failed nodes' replacement · Service classification is done based on the type of messages · At broker level
M/M/1	(Leu et al., 2014)	SPT	Service rate allocation is done on the basis of priority; provides higher priority to messages with SPT · Service classification is done based on service priority (High or Low) · At application and network Layer
M/M/1	(Abdullah and Yang, 2013b)	NA	Service rate allocation is done based on the priority of the services · Service classification is done based on service priority (High or Low) · At broker level
M/M/m/N	(Narman et al., 2016)	Dedicated server	If high priority requests increase then dynamically allocates resources to them, which was previously allocated to low priority requests · Service classification is done based on the priority · At application layer
M/G/m/m	(Edemacu and Bulega, 2014)	Time division	Periodic channel allocation is performed using fixed or dynamic periods for grouped traffic · Service classification is done based on the source of traffic · At network layer
Empirical model	(Xing et al., 2017)	Fair allocation	Achieves fairness maximization by batch allocation of resources while using resource utilization as a parameter · No service classification is required · At application layer
Empirical model	(Paulraj et al., 2018)	Round Robin	Managing resource starvation and effective resource migration; the complete resource pool is categorized using different combinations of the resources. Requests, based on the requirements, are migrated to these categorized resources. · No prior service classification is required · At application layer and network layer
Empirical model	(Bharti and Pattanaik, 2016)	Task preprocessing based	Proper energy monitoring using a weight factor to select a sensing node for a task, which is parameterized by the location, current energy, and task requirements · No service classification is required · At sensing layer
SDN	(Thubert et al., 2015)	TSCH	Cell selection from slotted channel is done using the path computation element control protocol · No service classification is required · At network layer
Empirical model	(Domingo-Prieto et al., 2016)	PID	Cell selection from slotted channel is done based on Proportional, Integral and Derivative (PID) control algorithm and also re-scheduling is done under performing cells · No service classification is required · At network layer
Empirical model	(Kuo and Chou, 2015)	Fuzzy allocation	Fuzzy theory is used to provide weights to the possible allocation based on a fuzzy membership function and best case is chosen with maximum membership value · Service classification is done based on priority · At network layer
Factor graph	(Sohn and Lee, 2015)	Proximity based	Closely located IoT nodes are scheduled at different time intervals to reduce interference · No service classification is required · At sensing layer
MDP	(Leu et al., 2014)	Decision making at each layer	An optimal policy is derived by reward maximization, which uses several metrics as QoS parameters · Service classification based on delay sensitive and best effort parameters

55

Table 6 (*continued*)

| Model	References	Scheduling Criteria	Working Principle and Working Mechanism
PSO	(S and P, 2015)	Dynamic scheduling	· At application, network, and sensing layer Achieves an optimal solution by PSO using load and response time as a parameter · Service classification is done based on service priority (High or Low) · At application layer
Simulated annealing	(Moschakis and Karatza, 2015)	Least loaded cloud first	Initially starts with SQF then uses simulated annealing processes to find a better solution for scheduling · No service classification is required · At application layer
Empirical model	(Sarangi et al., 2018)	Best Fit	Deadline based decision making for incoming service requests minimizing the overall energy consumption; first scheduling is done using best fit approach if the required CPU cores are not available, then based on the deadline requirement, the execution time is estimated using Global or Local algorithms. · Service classification is done based on deadline · At application, network, and sensing layer
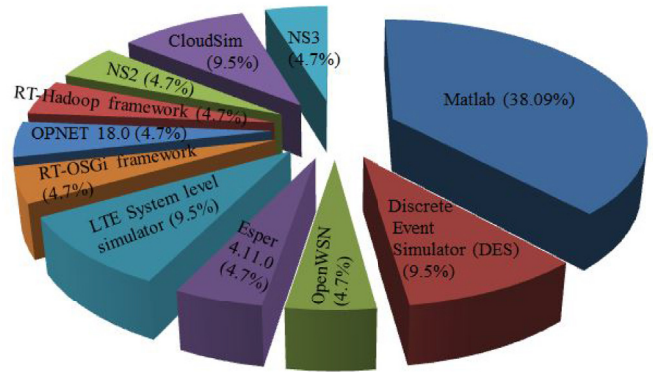
Fig. 12. Leading simulators for IoT scheduling.

a broker for each node. For simulation of the broker selection mechanism and the transmission data architecture in IoT networks, LEACH (Abdullah and Yang, 2013a) can be used.

Simulation of integrated environment of IoT and cloud, the CloudSim simulator can be used. Techniques like particle swarm optimization can easily be integrated with CloudSim as a new class. A distributed processing programming model like MapReduce can also be used for processing the large amount of data, generated by the IoT devices. IoTSim (Zeng et al., 2017), which is a combination of MapReduce and the CloudSim simulator, can be used to synthesize the IoT data processing with cloud resource management.

Fig. 12 shows different simulators used in the IoT environment to validate the scheduling algorithms where Table 7 matches the simulators with the metrics used in different scheduling algorithms, referred in this research article. From our analysis, we can see in most of the cases Matlab is used to simulate the necessary scheduling environment. But nowadays other efficient simulators such as OpenIoT (Soldatos et al., 2015), Cooja Simulator - Contiki for IoT (Khakimov et al., 2017), etc. can be used for this purpose.

6. Immediate research needs/directions

Though, a good number of researchers have worked hard to achieve optimal solutions regarding IoT resource scheduling. Still, there exist many crucial issues and challenges that are essential to be addressed in upcoming researches. We have identified such open issues that need faster attention.

6.1. Smart systems

In smart IoT eco-systems like smart city, smart home, or smart

Table 7
Mapping of simulators with the metrics.

Simulators	Considered Metrics
Matlab	Load, Energy, Response Time, Waiting Time, Throughput, Information Accuracy Blocking Probability, Latency, Traffic
Discrete Event Simulator (DES)	Service Cost, Drop Rate, Throughput, Load
Esper 4.11.0	Service Time, Fairness
NS-2	Waiting Time
NS-3	Energy
OPNET	Delay, Energy, Waiting Time, Load, Service Cost
RT-OSGi Framework	Response Time
RT-Hadoop Framework	Energy
CloudSim	Energy, Fairness
OpenWSN	Energy, Traffic
LTE System level simulator	Traffic, Blocking Probability, Load, Throughput, Co-channel interference

healthcare where automation of user requirement is done using various IoT devices. Smart systems consist of a large number of heterogeneous IoT resources for which proper scheduling standards are required. Also, middleware technology needs to be developed for handling influential concerns like energy utilization and congestion control. As per the literature (Naz et al., 2018; Shafie-Khah and Siano, 2018), the residential sector consumes more than 80% of overall electricity, therefore, optimization of home energy management is becoming a challenging task from the researcher's viewpoint. Hence, to reduce the unnecessary energy consumption, it is important to schedule the appliances in such way that overall load can be balanced as well as their operations are prioritized as per the consumers' demand. On the other hand, in IoT-enabled remote healthcare monitoring system (Chen et al., 2011; Patel and Wang, 2010), all the monitoring devices have different sensors which are connected to a local data processing unit through a shared channel with fixed bandwidth. Therefore, channel congestion and channel interference are of great concern in this case as the vital health data should be distortion free. Hence, immediate research regarding interference mitigation among sensors must be carried out so that there is no overlapping of data while going through the shared channel.

6.2. QoS

QoS requirements precisely affect the QoE (Quality of Experience) of the customer. It is a higher level objective which can be achieved by setting lower level objectives like better energy management, throughput maximization, blocking probability reduction, etc. New work can be done which will incorporate energy-based threshold to resolve non-uniform distribution of energy in IoT nodes and instead of randomized technique, better broker selection can be done based on energy available at each node (broker candidate) or resources available (Abdullah and Yang, 2013a, 2014). Research concern is required in the context of optimal path selection algorithm for broker to broker communication. For better management of IoT composite services (Ming and Yan, 2013), a backtracking scheduling scheme is required which will effectively deal with context switching between atomic processes of composite services and also deal with the uncertainty of QoS. For better QoS in a highly dense IoT network, proper standardization of communication is required to reduce the co-channel interference. However, all these objectives are typical network requirements. Therefore, relevant solutions for critical QoS requirements of multiple IoT application domains is a subject of research.

6.3. Heterogeneity/big data

To handle tremendous heterogeneous data is a big issue/challenge in today's scenario (Sivarajah et al., 2017). Upon that, there are also limitations in terms of standardization, data format (heterogeneous), big data management (scalability), and operational of IoT (Fang et al., 2015). Therefore, there is a need of new seamless scheduling method that can deal well with these issues by proper standardization of data storage and processing by providing efficient access to the storage resources/capacity and computing power.

6.4. IoTCloud environment

IoT is lacking in storing and managing the huge amount of data or Big data, because of its limited storage capacity, and limited processing and energy resources. Therefore, it is an intelligent idea to integrate cloud with IoT and build an IoTCloud platform (Hossain et al., 2017; Cai et al., 2017; Díaz et al., 2016) to create large scale, complex, and distributed eco-system as cloud computing is a source of virtually unlimited storage and computational capabilities (Botta et al., 2016). In case of IoTCloud integrated environment, job scheduling is done for mapping the incoming jobs with the available VMs (Moschakis and Karatza, 2015). New parameters can be added to the existing pool of

deciding parameters for this mapping. One approach will be applying a reliability based scheduling of virtual machines by quantifying the reliability of each VM. Hence, for IoTCloud platform new resource scheduling techniques need to be evolved that will optimize the resource utilization for storage and processing of IoT data as well as minimize the expenses for accessing location dependent IoT resources and other sensor nodes readings.

6.5. Fog computing

Fog computing (Hu et al., 2017) can be integrated with IoT for some specific IoT applications, such as, video streaming, gaming, moving vehicles, etc. which require low network latency. Fog computing serves applications with extensively distributed deployments. It extends the cloud computing by importing the cloud resources closer to the IoT devices (Aazam and Huh, 2014; Sarkar et al., 2015; Mahmud and Buyya, 2016). So, when IoT integrates fog computing, new resource management schemes need to be proposed especially for smart communication of IoT applications. The simulator iFogSim (Gupta et al., 2016) in this context can be used to model and simulate the resource management and application scheduling across the edge nodes in an IoT environment.

6.6. Middleware

In any service oriented system, a middleware can simplify the development process through the integration between heterogeneous computing and communication resources. It also supports the interoperability within a diverse range of application domains and services. Based on several perspectives, in the recent years, a good number of proposals (Azzara et al., 2013; Bandyopadhyay et al., 2011; Delicato et al., 2013; Ngu et al., 2017) have already been suggested, but these are basically WSN centric. Therefore, to meet the acceptable level of QoS and QoE standards for IoT customers, there is a need of management middleware for better monitoring, allocation, and deallocation of IoT resources (Razzaque et al., 2016). In this regard, potential scheduling approaches are required to be developed for middleware in order to manage the resources between the application and the network communication layer.

6.7. Service level agreement (SLA)

There is a need of a dynamic IoT scheduling approach to achieve the QoS requirements desired by IoT users in terms of SLA. As the traditional schemes to handle SLA are limited to a predefined service quality granted for a fixed cost (Mubeen et al., 2017; Grubitzsch et al., 2017). So, an effective scheduling technique should be proposed which will use minimization of SLA violation as an objective function in order to avoid re-negotiation or service termination. This will further improve the overall performance as well as the reputation of the service provider.

6.8. Real-time data processing

Proper allocation of resources for real-time data processing is an open challenge of IoT system. Available proposals are still lacking to fulfil the demand of resources while performing the tasks involved in a typical workflow of data stream processing or real time processing. Therefore, resource allocation solutions for such concerns need to be designed for fair assignment of data stream process tasks.

There are several other issues in IoT device scheduling like:

· Proper deployment techniques for IoT devices which assures low co-channel interference need to be developed.
· More sophisticated techniques are required for co-existing, dependent devices for management of their resources, channels, and intermediate communications.

· Formulation of a deterministic approach for an IoT network while ensuring the quality of service.
· Dependent things' task classification based on special and temporal requirements.
· Streaming the data from application layer in form of parameters, then using this data as an input to the deterministic model for resource scheduling.
· An energy utilization of mathematical programming model needs to be done for maximization problem.
· Allocation of heterogeneous virtual machines based on user request for IoT services.
· In case of prioritized traffic, proper buffer management for reduction of blocking probability.
· In a job arrival scenario where heterogeneous jobs are arriving, heterogeneity needs to be considered as a parameter and a proper queueing scheme should be proposed for these heterogeneous job requests.
· Some green computing techniques can be used (green computing + IoT) in scheduling purpose to achieve energy efficient Green Internet of Things (Yaacoub et al., 2012).

7. Conclusion and future work

In this survey study, the outcomes have been investigated in several ways; such as, classification of IoT resources, IoT resource management activities, scheduling process in IoT environment, and most importantly a meticulous review of several resource scheduling schemes in IoT environment based on three elementary concerns, i.e. QoS, IoT architecture, and IoT network infrastructure. Subclassification of resource scheduling under each category has also been done in this work. Many proposals have focused on this topic while considering distinct objectives. Most of such proposals have been reviewed and summarized in terms of their prominent objectives and metrics used. Possible scope of improvements for each proposal has been observed and provided as remarks. To achieve broader objectives like QoS, optimization of the metrics such as load, energy efficiency, throughput, information accuracy, traffic management, resource utilization, response time, delay, waiting time, blocking probability, etc. need to be carried out.

The noteworthy observation of this study is the specific parameter based analysis. It shows that energy efficiency, load management, and throughput are the most explored parameters in IoT resource scheduling. A detailed comparison based on modeling strategy and working principle of different IoT scheduling techniques have been carried out. Mapping of metrics (involved in this study) with the simulation tools has also been done. Subsequently, several research gaps have been identified that require urgent attention in order to solve the challenges of IoT resource management. Recent researches are lacking to have a keen interest in developing a generic scheme which covers all the aspects of IoT scheduling. Thus, considering the heterogeneity and dynamic nature of IoT objects, researchers need to tackle several parameters ensuring QoS and performance of the system. There are several technologies, such as, cloud computing, fog computing, green computing, and edge computing that need to be integrated with IoT to achieve higher level objectives with efficiency. Therefore, it becomes essential to incorporate adaptability in such techniques for multi objective optimization in IoT system domain.

In the immediate future, a detailed simulation based analysis of various scheduling methods will be done and an efficient scheduling scheme will be proposed that can maintain the trade-off between all metrics while implementing different IoT applications to get the best performance from an IoT system. Also, we intend to investigate service provisioning in IoT environment. The ultimate aim is to develop an energy efficient novel scheduling approach for optimization of resource consumption and balancing load in integrated cloud computing and IoT environment for enhancement in overall performance. The state information about energy consumption and resource utilization in IoT nodes and information about task processing are sent to scheduler periodically for evaluation and as a reference. Thus, a balanced resource management scheme will be implemented for a broader IoT ecosystem in our next research.

References

Aazam, M., Huh, E.-N., 2014. Fog computing and smart gateway based communication for cloud of things. In: Future Internet of Things and Cloud (FiCloud), 2014 International Conference on. IEEE, pp. 464–470.

Abdullah, S., Yang, K., 2013a. An energy-efficient message scheduling algorithm in internet of things environment. In: Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International. IEEE, pp. 311–316.

Abdullah, S., Yang, K., 2013b. A qos aware message scheduling algorithm in internet of things environment. In: Online Conference on Green Communications (GreenCom), 2013 IEEE. IEEE, pp. 175–180.

Abdullah, S., Yang, K., 2014. An energy efficient message scheduling algorithm considering node failure in iot environment. Wireless Pers. Commun. 79, 1815–1835.

Adan, I., Resing, J., 2015. Queueing Systems. Eindhoven University of Technology, Department of Mathematics and Computing Science.

Al-Fagih, A.E., Al-Turjman, F.M., Alsalih, W.M., Hassanein, H.S., 2013. A priced public sensing framework for heterogeneous iot architectures. IEEE Trans. Emerg. Top. Comput. 1, 133–147.

Anastasi, G., Conti, M., Di Francesco, M., Passarella, A., 2009. Energy conservation in wireless sensor networks: a survey. Ad Hoc Netw. 7, 537–568.

Ashton, K., 2009. That 'internet of things' thing. RFiD J. 22, 97–114.

Atzori, L., Iera, A., Morabito, G., 2010. The internet of things: a survey. Comput. Network. 54, 2787–2805.

Azzara, A., Bocchino, S., Pagano, P., Pellerano, G., Petracca, M., 2013. Middleware solutions in wsn: the iot oriented approach in the icsi project. In: Software, Telecommunications and Computer Networks (SoftCOM), 2013 21st International Conference on. IEEE, pp. 1–6.

Bandyopadhyay, D., Sen, J., 2011. Internet of things: applications and challenges in technology and standardization. Wireless Pers. Commun. 58, 49–69.

Bandyopadhyay, S., Sengupta, M., Maiti, S., Dutta, S., 2011. A survey of middleware for internet of things. In: Recent Trends in Wireless and Mobile Networks. Springer, pp. 288–296.

Bharti, S., Pattanaik, K., 2016. Task requirement aware pre-processing and scheduling for iot sensory environments. Ad Hoc Netw. 50, 102–114.

Botta, A., De Donato, W., Persico, V., Pescapé, A., 2016. Integration of cloud computing and internet of things: a survey. Future Generat. Comput. Syst. 56, 684–700.

Buyya, R., Dastjerdi, A.V., 2016. Internet of Things: Principles and Paradigms. Elsevier.

Cai, H., Xu, B., Jiang, L., Vasilakos, A.V., 2017. Iot-based big data storage systems in cloud computing: perspectives and challenges. IEEE Internet Things J. 4, 75–87.

Castellani, A.P., Bui, N., Casari, P., Rossi, M., Shelby, Z., Zorzi, M., 2010. Architecture and protocols for the internet of things: a case study. In: Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010 8th IEEE International Conference on. IEEE, pp. 678–683.

Chen, M., Gonzalez, S., Vasilakos, A., Cao, H., Leung, V.C., 2011. Body area networks: a survey. Mobile Network. Appl. 16, 171–193.

Claudio, F., 2013. Master Degree Network Management and Qos Provisioning Scheduling. http://claudiofiandrino.altervista.org/Master_degree/Network_management_and_QoS_provisioning/scheduling.pdf [Online; accessed 2017-03-25].

Crosby, G.V., Vafa, F., 2013. Wireless sensor networks and lte-a network convergence. In: Local Computer Networks (LCN), 2013 IEEE 38th Conference on. IEEE, pp. 731–734.

Delicato, F.C., Pires, P.F., Batista, T., 2013. Middleware Solutions for the Internet of Things. Springer.

Delimitrou, C., Kozyrakis, C., 2013. Qos-aware scheduling in heterogeneous datacenters with paragon. ACM Trans. Comput. Syst. 31, 12.

Díaz, M., Martín, C., Rubio, B., 2016. State-of-the-art, challenges, and open issues in the integration of internet of things and cloud computing. J. Netw. Comput. Appl. 67, 99–117.

Domingo-Prieto, M., Chang, T., Vilajosana, X., Watteyne, T., 2016. Distributed pid-based scheduling for 6tisch networks. IEEE Commun. Lett. 20, 1006–1009.

Edemacu, K., Bulega, T., 2014. Resource sharing between m2m and h2h traffic under time-controlled scheduling scheme in lte networks. In: Telecommunication Systems Services and Applications (TSSA), 2014 8th International Conference on. IEEE, pp. 1–6.

Eswaran, S.P., Ariharan, V., Bapat, J., 2014. Event driven opportunistic communication enabler for smart city. In: Next Generation Mobile Apps, Services and Technologies (NGMAST), 2014 Eighth International Conference on. IEEE, pp. 313–319.

Eswaran, S.P., Bapat, J., 2015. Service centric markov based spectrum sharing for internet of things (iot). In: Region 10 Symposium (TENSYMP), 2015 IEEE. IEEE, pp. 9–12.

Fang, S., Xu, L., Zhu, Y., Liu, Y., Liu, Z., Pei, H., Yan, J., Zhang, H., 2015. An integrated information system for snowmelt flood early-warning based on internet of things. Inf. Syst. Front 17, 321–335.

Geng, X., Luo, A., Sun, Z., Cheng, Y., 2012. Markov chains based dynamic bandwidth allocation in diffserv network. IEEE Commun. Lett. 16, 1711–1714.

Givan, B., Parr, R., 2001. An Introduction to Markov Decision Processes. Purdue University.

Grubitzsch, P., Braun, I., Fichtl, H., Springer, T., Hara, T., Schill, A., 2017. Ml-sla: multi-level service level agreements for highly flexible iot services. In: Internet of Things (ICIOT), 2017 IEEE International Congress on. IEEE, pp. 113–120.

Gupta, H., Dastjerdi, A.V., Ghosh, S.K., Buyya, R., 2016. Ifogsim: a Toolkit for Modeling and Simulation of Resource Management Techniques in Internet of Things, Edge and Fog Computing Environments. arXiv preprint arXiv:1606.02007.

Hossain, M.S., Xu, C., Li, Y., Pathan, A.-S.K., Bilbao, J., Zeng, W., El Saddik, A., 2017. Impact of next-generation mobile technologies on iot-cloud convergence. IEEE Commun. Mag. 55, 18–19.

Hu, P., Dhelim, S., Ning, H., Qiu, T., 15 November 2017. Survey on fog computing: architecture, key technologies, applications and open issues. J. Netw. Comput. Appl. 98, 27–42.

Kevin, F., 2016. Lte and the Internet of Things. http://www.3gpp.org/news-events/3gpp-news/1607-i%20ot [Online; accessed 2017-03-28].

Khakimov, A., Muthanna, A., Kirichek, R., Koucheryavy, A., Muthanna, M.S.A., 2017. Investigation of methods for remote control iot-devices based on cloud platforms and different interaction protocols. In: Young Researchers in Electrical and Electronic Engineering (EIConRus), 2017 IEEE Conference of Russian. IEEE, pp. 160–163.

Kim, H.-W., Park, J.H., Jeong, Y.-S., 2016. Efficient resource management scheme for storage processing in cloud infrastructure with internet of things. Wireless Pers. Commun. 91, 1635–1651.

Kim, W., 2016. Adaptive resource scheduling for dual connectivity in heterogeneous iot cellular networks. Int. J. Distributed Sens. Netw. 12 (4).

Kim, W., Andreas, J.K., Di Felice, M., Gerla, M., 2010. Cognitive multi-radio mesh networks on ism bands: a cross-layer architecture. In: Performance Computing and Communications Conference (IPCCC), 2010 IEEE 29th International. IEEE, pp. 34–41.

Kuo, Y.-W., Chou, L.-D., 2015. Power saving scheduling scheme for internet of things over lte/lte-advanced networks. Mobile Inf. Syst. 2015, 11 971538.

Leu, J.-S., Chen, C.-F., Hsu, K.-C., 2014. Improving heterogeneous soa-based iot message stability by shortest processing time scheduling. IEEE Trans. Serv. Comput. 7, 575–585.

Li, L., Li, S., Zhao, S., 2014. Qos-aware scheduling of services-oriented internet of things. IEEE Trans. Ind. Inf. 10, 1497–1505.

Li, S., Da Xu, L., Zhao, S., 2015. The internet of things: a survey. Inf. Syst. Front 17, 243–259.

Li, S., Xu, L., Wang, X., Wang, J., 2012a. Integration of hybrid wireless networks in cloud services oriented enterprise information systems. Enterprise Inf. Syst. 6, 165–187.

Li, W.-X., Xu, J., Jiang, H., 2012b. Queuing states analysis on a hybrid scheduling strategies for heterogeneous traffics in iot. In: Computer Science & Service System (CSSS), 2012 International Conference on. IEEE, pp. 1007–1008.

Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H., Zhao, W., Oct. 2017. A survey on internet of things: architecture, enabling technologies, security and privacy, and applications. IEEE Internet Things J. 4 (5), 1125–1142 2327-4662.

Liu, W., Nishio, T., Shinkuma, R., Takahashi, T., 2014a. Adaptive resource discovery in mobile cloud computing. Comput. Commun. 50, 119–129.

Liu, Y., Chen, Z., Lv, X., Han, F., 2014b. Multiple layer design for mass data transmission against channel congestion in iot. Int. J. Commun. Syst. 27, 1126–1146.

Lublin, U., Feitelson, D.G., 2003. The workload on parallel supercomputers: modeling the characteristics of rigid jobs. J. Parallel Distr. Comput. 63, 1105–1122.

MacLennan, E., Van Belle, J.-P., 2014. Factors affecting the organizational adoption of service-oriented architecture (soa). Inf. Syst. E Bus. Manag. 12, 71–100.

Mahmud, R., Buyya, R., 2016. Fog Computing: a Taxonomy, Survey and Future Directions. arXiv preprint arXiv:1611.05539.

Mainetti, L., Patrono, L., Vilei, A., 2011. Evolution of wireless sensor networks towards the internet of things: a survey. In: Software, Telecommunications and Computer Networks (SoftCOM), 2011 19th International Conference on. IEEE, pp. 1–6.

Marini, F., Walczak, B., 2015. Particle swarm optimization (pso). a tutorial. Chemometr. Intell. Lab. Syst. 149, 153–165.

Ming, Z., Yan, M., 2013. Qos-aware computational method for iot composite service. J. China Univ. Posts Telecommun. 20, 35–39.

Moschakis, I.A., Karatza, H.D., 2015. Towards scheduling for internet-of-things applications on clouds: a simulated annealing approach. Concurrency Comput. Pract. Ex. 27, 1886–1899.

Mubeen, S., Asadollah, S.A., Papadopoulos, A.V., Ashjaei, M., Pei-Breivold, H., Behnam, M., 2017. Management of service level agreements for cloud services in iot: a systematic mapping study. IEEE Access 6 2169-3536.

Munir, A., Gordon-Ross, A., 2011. Markov modeling of fault-tolerant wireless sensor networks. In: Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on. IEEE, pp. 1–6.

Nace, D., Pióro, M., 2008. Max-min fairness and its applications to routing and load-balancing in communication networks: a tutorial. IEEE Commun. Surv. Tutorials 10.

Narman, H.S., Hossain, M.S., Atiquzzaman, M., Shen, H., 2016. Scheduling internet of things applications in cloud computing. Annals Telecommun. 1–15.

Naz, M., Iqbal, Z., Javaid, N., Khan, Z.A., Abdul, W., Almogren, A., Alamri, A., 2018. Efficient power scheduling in smart homes using hybrid grey wolf differential evolution optimization technique with real time and critical peak pricing schemes. Energies 11, 384.

Ngu, A.H., Gutierrez, M., Metsis, V., Nepal, S., Sheng, Q.Z., 2017. Iot middleware: a survey on issues and enabling technologies. IEEE Internet Things J. 4, 1–20.

NOKIA, 2016. Lte Evolution for Iot Connectivity. http://resources.alcatel-lucent.com/asset/200178 [Online; accessed 2017-03-25].

Patel, M., Wang, J., 2010. Applications, challenges, and prospective in emerging body area networking technologies. IEEE Wirel. Commun. 17.

Paulraj, G.J.L., Francis, S.A.J., Peter, J.D., Jebadurai, I.J., 2018. Resource-aware virtual machine migration in iot cloud. Future Generat. Comput. Syst. 85, 173–183.

Rahman, A., Wang, C., 2016. Resource Discovery in the Internet of Things. http://www.interdigital.com/download/5772781ce3b965adf50018df [Online; accessed 2017-03-28].

Razzaque, M.A., Milojevic-Jevric, M., Palade, A., Clarke, S., 2016. Middleware for internet of things: a survey. IEEE Internet Things J. 3, 70–95.

Rowland, C., Goodman, E., Charlier, M., Light, A., Lui, A., 2015. Designing Connected Products: UX for the Consumer Internet of Things. O'Reilly Media, Inc.

S., K., P, J.P., April 2015. Qos aware resource scheduling in internet of things-cloud environment. Int. J. Sci. Eng. Res. 6 (4), 294 2229-5518.

Sarangi, S.R., Goel, S., Singh, B., 2018. Energy Efficient Scheduling in Iot Networks.

Sarkar, S., Chatterjee, S., Misra, S., 01 October 2015. Assessment of the suitability of fog computing in the context of internet of things. IEEE Trans. Cloud Comput. 6 (1), 46–59 2168-7161, Jan.-March 2018.

Shafie-Khah, M., Siano, P., 2018. A stochastic home energy management system considering satisfaction cost and response fatigue. IEEE Trans. Ind. Inf. 14, 629–638.

Sharma, R., Kumar, N., 2015. Qos-alert Markov chain based scheduling scheme in internet of things. In: Globecom Workshops (GC Wkshps), 2015 IEEE. IEEE, pp. 1–6.

Sharma, R., Kumar, N., Gowda, N.B., Srinivas, T., 2015. Probabilistic prediction based scheduling for delay sensitive traffic in internet of things. Proced. Comput. Sci. 52, 90–97.

Singh, A., Viniotis, Y., 2017. Resource allocation for iot applications in cloud environments. In: Computing, Networking and Communications (ICNC), 2017 International Conference on. IEEE, pp. 719–723.

Sivarajah, U., Kamal, M.M., Irani, Z., Weerakkody, V., 2017. Critical analysis of big data challenges and analytical methods. J. Bus. Res. 70, 263–286.

Sohn, I., Lee, S.H., 2015. Distributed scheduling for coexistence of iot wireless devices. In: Information and Communication Technology Convergence (ICTC), 2015 International Conference on. IEEE, pp. 680–682.

Soldatos, J., Kefalakis, N., Hauswirth, M., Serrano, M., Calbimonte, J.-P., Riahi, M., Aberer, K., Jayaraman, P.P., Zaslavsky, A., Žarko, I.P., et al., 2015. Openiot: open source internet-of-things in the cloud. In: Interoperability and Open-source Solutions for the Internet of Things. Springer, pp. 13–25.

Thubert, P., Palattella, M.R., Engel, T., 2015. 6tisch centralized scheduling: when sdn meet iot. In: Standards for Communications and Networking (CSCN), 2015 IEEE Conference on. IEEE, pp. 42–47.

Tyagi, S., Kumar, N., 2013. A systematic review on clustering and routing techniques based upon leach protocol for wireless sensor networks. J. Netw. Comput. Appl. 36, 623–645.

Vassaki, S., Georgakopoulos, A., Miatton, F., Tsagkaris, K., Demestichas, P., 2016. Interference and qos aware channel segregation for heterogeneous networks: a preliminary study. In: Networks and Communications (EuCNC), 2016 European Conference on. IEEE, pp. 195–199.

Vermesan, O., Friess, P., Guillemin, P., Gusmeroli, S., Sundmaeker, H., Bassi, A., Jubert, I.S., Mazura, M., Harrison, M., Eisenhauer, M., et al., 2011. Internet of things strategic research roadmap. Internet Things-Global Technol. Soci. Trends 1, 9–52.

Watteyne, T., Vilajosana, X., Kerkez, B., Chraim, F., Weekly, K., Wang, Q., Glaser, S., Pister, K., 2012. Openwsn: a standards-based low-power wireless development environment. Trans. Emerg. Telecommun. Technol. 23, 480–493.

Whitmore, A., Agarwal, A., Da Xu, L., 2015. The internet of things—a survey of topics and trends. Inf. Syst. Front 17, 261–274.

Xing, G., Xu, X., Xiang, H., Xue, S., Ji, S., Yang, J., 2017. Fair energy-efficient virtual machine scheduling for internet of things applications in cloud environment. Int. J. Distributed Sens. Netw. 13 1550147717694890.

Yaacoub, E., Kadri, A., Abu-Dayya, A., 2012. Cooperative wireless sensor networks for green internet of things. In: Proceedings of the 8h ACM Symposium on QoS and Security for Wireless and Mobile Networks. ACM, pp. 79–80.

Zarrinkoub, H., 2014. Understanding LTE with MATLAB: from Mathematical Modeling to Simulation and Prototyping. John Wiley & Sons.

Zeng, X., Garg, S.K., Strazdins, P., Jayaraman, P.P., Georgakopoulos, D., Ranjan, R., 2017. Iotsim: a simulator for analysing iot applications. J. Syst. Architect. 72, 93–107.

Zhang, J., Song, G., Qiao, G., Li, Z., Wang, A., 2012. A wireless sensor network system with a jumping node for unfriendly environments. Int. J. Distributed Sens. Netw. 2012.

Zhou, H., 2013. The Internet of Things in the Cloud. CRC Press, Taylor & Francis Group.,.

Abishi Chowdhury has done her B. E. from University Institute of Technology, West Bengal, India and M. Tech. from National Institute of Technical Teachers' Training and Research, Bhopal, Madhya Pradesh, India. Currently, she is pursuing Ph. D. from Visvesvaraya National Institute of Technology, Nagpur, India. Her research interest includes Cloud Computing, Cloud resource scheduling and Internet of Things (IoT) and she has publications in similar areas.

Shital A. Raut has done B. E. in Comp. Tech, M.E. in Computer Science & Engineering, and currently attached with Department of Computer Science & Engineering as Asst. Professor, Visvesvaraya National Institute of Technology, Nagpur, India. She has done her Ph. D. from Department of Computer Science and Engineering, Visvesvaraya National Institute of Technology, Nagpur, India. Her current research interest includes bioinformatics, algorithms, Data Mining and Warehousing, Business Information Systems and Internet of Things and she has publications in the similar areas.