Fog computing scheduling algorithm for smart city

Ahmad Mohammad Alsmadi[1], Roba Mahmoud Ali Aloglah[2], Nisrein Jamal Sanad Abu-darwish[3], Ahmad Al Smadi[4], Muneerah Alshabanah[5], Daniah Alrajhi[6], Hanouf Alkhaldi[7], Mutasem K. Alsmadi[8]
[1,2,3]Department of Computer Science, Al-Balqa Applied University, Ajloun University College, Jordan
[4]School of Artificial Intelligence, Xidian University, China
[5,6,7,8]Department of MIS, College of Applied Studies and Community Service, Imam Abdulrahman Bin Faisal University, Dammam, Saudi Arabia

Article Info

Article history:

Received Sep 23, 2020
Revised Dec 11, 2020
Accepted Dec 15, 2020

Keywords:

Fog computing
Fog simulator
Smart city
Task scheduling

ABSTRACT

With the advent of the number of smart devices across the globe, increasing the number of users using the Internet. The main aim of the fog computing (FC) paradigm is to connect huge number of smart objects (billions of objects) that can make a bright future for smart cities. Due to the large deployments of smart devices, devices are expected to generate huge amounts of data and forward the data through the Internet. FC also refers to an edge computing framework that mitigates the issue by applying the process of knowledge discovery using a data analysis approach to the edges. Thus, the FC approaches can work together with the internet of things (IoT) world, which can build a sustainable infrastructure for smart cities. In this paper, we propose a scheduling algorithm namely the weighted round-robin (WRR) scheduling algorithm to execute the task from one fog node (FN) to another fog node to the cloud. Firstly, a fog simulator is used with the emergent concept of FC to design IoT infrastructure for smart cities. Then, spanning-tree routing (STP) protocol is used for data collection and routing. Further, 5G networks are proposed to establish fast transmission and communication between users. Finally, the performance of our proposed system is evaluated in terms of response time, latency, and amount of data used.

Corresponding Author:

Ahmad Mohammad Alsmadi
Department of Computer Science
Al-Balqa Applied University, Ajloun University College, Jordan
Email: amhs1966@bau.edu.jo

1. INTRODUCTION

Today, many IoT applications that leverage pervasive connectivity and cloud computing are permitting the initiatives of the smart city all over the world [1-10]. These applications based on cloud computing propose new capabilities including monitoring system ability, control and manage devices remotely, and to make new actionable information and insights to keep massive access streams of real-time data [11]. Over the past few decades, only 1 out of 10 people can live in urban areas since we were in the 21st century. In 2008, the population of the world crossed the line of being more than 50% urban and it is expected to reach 75% by 2050. While this could cause rapid urbanization, fuel economic growth will highly stress infrastructure in smart cities that provide high demands for energy, water, transportation, housing, and healthcare applications [12, 13]. Due to finite resources, demographic shifts, limited budgets, climate change concerns, it is required to use emerging technologies effectively to obtain high efficiency and improve the quality of life in urban areas. Most essentially, it helps to build environmentally and economically sustainable cities.

According to information and communication technologies (ICT), there is no tipping point after which a city can be termed as a “Smart”. Precise solutions could easily make the city “smart” and improve the urban city performance, the smartness is due to the optimization of natural resources, managing, and monitoring infrastructure, and improving cost-effectiveness that keeps cities running smoothly. A study in [14] reported from a BBC news article that IBM had more than 2,500 smart city projects around the world in 2013. The company had worked with few government sectors on smart city initiatives such as Dubuque (water), Singapore (traffic), Dublin (parking), Rio (traffic), California (traffic), and Stockholm (traffic), etc. There have been also other smart city pilot projects led initiatives like those in the cities of “New York” (sewage/rainwater), and London (weather, pollution, river levels) where few local governments opened up city data to the public [15]. Recently, the infrastructure for cities is being smarted [16, 17], as retrofitting smart technologies into the previous infrastructure has been growing around the world such as New York city, Singapore city, and Mexico, etc.

Fog computing (FC) is a term that extended cloud computing, the fog layer is located between the cloud service and local resource, in which some operations such as, network services, storage, and computing are situated between these two areas, normally at local area network (LAN) level. Figure 1 illustrates the FC application layer [18]. In an FC environment, an enormous number of FNs will be geographically distributed to service local resources in their particular regions. Each of these FNs will be able to leverage tasks for clients and take part in service orchestration with underlying computing resources in the area. To deal with diverse IoT smart objects, the FC framework has been proposed; however, it is expected to solve open issues such as energy efficiency, mobility, and latency. Cloud computing solutions could not be applied to future applications, including smart buildings, smart cities, etc. Therefore, FC extends cloud computing to use a massive volume of data, and to report real-time constraints in the network. Besides, the framework for FC can give efficient ways to overcome various limitations of the previous network frameworks [19]. Another advantage is that the FC layer between the cloud and IoT devices has valuable functionalities [20].

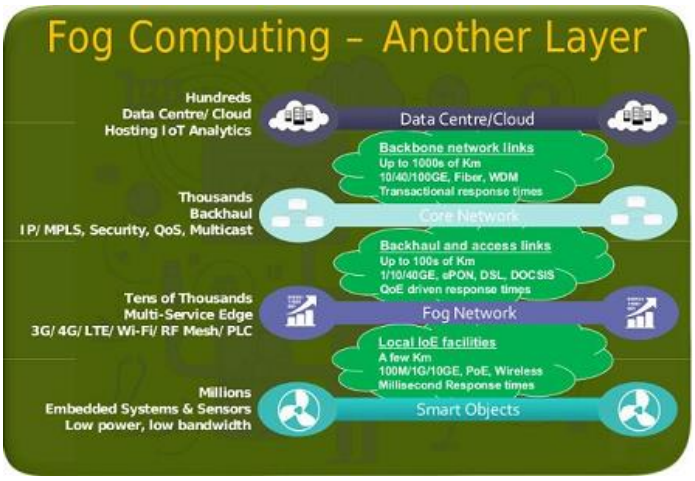


Figure 1. FC application layer [18]

a. Constraints on latency

FC architectures can provide an opportunity for data analytics and big data real-time processing over the network edge. This benefit will permit the development of time-sensitive and delay-sensitive applications, and this will deal with the strict requirements of future applications for smart cities. In this way, the FC paradigm assists current researchers to decrease the delay in deployment of most service provisioning applications [21], moreover, the paradigm of FC has been presented to support the latency-sensitive computational demand [22].

b. Constraints on security

Previous studies for security solutions were demonstrated for protecting huge enterprise data centers and networks by using perimeter-based protections. Generally, these types of services for security solutions are not adequate to address the new challenges in emerging technology [23, 24]. In [25], the authors proposed a new FC paradigm that allowed various security functions (distributed monitoring of malware) for the diverse IoT devices to recompense the limited security of these devices. They benefited from the local information collection to deal with attacks and threats promptly.

c. Constraints of network bandwidth

For implementing smart city scenarios, centralized architectures are not applicable because sending all the data gathered by smart devices to the cloud layer needs a huge amount of network bandwidth, and also existing storage systems utilize an enormous amount of network bandwidth [26]. Hence, the FC paradigm allows local data analytics and processing operations, which considerably decreases the quantity of data that needs to be forwarded to the cloud system [27].

d. Constraints of resource of IoT devices

Smart devices utilize a limited resource such as the battery, computational power, memory capacity, and storage. These smart objects are not practicable to depend on these devices only to satisfy all the required computational operations. In such cases, FNs can carry out these IoT devices computational tasks despite resource-constrained IoT devices. However, FNs can minimize device complexity, energy consumption, and deployment costs [26].

e. Constraints of high dynamicity

Mobility is one of the essential requirements which need to be considered for smart city scenarios since a huge amount of resources will be demanded simultaneously by several devices at various locations. In a large-scale deployment, diverse devices can yield serious threats for addressing seamless mobility in heterogeneous environments. This is because devices are mobile and accessing the medium constantly by various technologies. To limit such mobility issues, the FC framework has been designed to render services and resources over the edge of the network to handle the handover problem effectively [28].

f. Distributed data analysis constraints

Firstly, smart devices can forward their samples of data to local FNs. Next detection of anomaly and monitoring operations will perform in a distributed manner. If malicious activities or anomalous behaviors are noticed in the data, a more rapid response can be reached. Accordingly, several malfunctions in smart objects can be detected and transmissions of wrong information can be prevented at the appropriate time. Hence local FNs can improve network reliability [29].

g. Decisions and local autonomous operations

As smart devices allow decisions and autonomous operations locally, this decreases the amount of data on-demand to enhance network reliability. FC applications in the smart city are the following [30]:

- Smart homes (home security, fire detection, and temperature control).
- Weather and water system (weather condition, and water quality control and leakage detection).
- Transportation and vehicular traffic applications (traffic reduction, assisted driving, and travel scheduling).
- Environmental control systems (energy efficiency monitoring, and renewable energy usage).
- Surveillance systems (violence detection and public place monitoring).

This paper concentrate on the FC environment task scheduling problem that handles the challenge of the growing need for computational resources by smart users to work professionally on a huge number of tasks. The problem of task scheduling optimally regulates the assignment of various submitted tasks to be performed using the minimum number of resources of FC such as a smaller amount of memory and in the shortest CPU time of execution. Consequently, the users can attain a faster time of execution with the minimum cost. In this work, the major contributions are following:

- Firstly, we deploy the network with several FNs. Next, find the optimal FN (i.e. fog server) and edge node for executing the user tasks based on the available processing capacity and the remaining node energy.
- Then to perform task execution, the scheduling algorithm called WRR Scheduling is proposed for executing the task to another FN to the cloud.
- FC network protocol uses a spanning-tree protocol (STP) for data collection and routing. We implement to allow autonomous MANO functionalities for applications of smart cities in 5G (5th generation) networks.
- We implemented this network model using the NS3 simulator and ifog simulators.
- Finally, evaluate the performance of the proposed system with display the results for the amount of data used, latency, and response time.

The following sections of this paper are structured as follows. The second section discusses the diverse approaches proposed for task scheduling in the FC environment and summaries this work contribution presented in this section (section 1). In the third section, the problem statement is organized. Section 4 demonstrates the proposed task scheduling and optimal nodes selection in the FC environment. We present a network design and performance evaluation of our proposed system with appropriate results for certain constraints in FC in section 5. Section 6 is the conclusion, also directions for future research are mentioned.

2. RELATED WORK

There are some studies on task scheduling in FC. In the computation latency tradeoff and energy consumption problem, when assigning the jobs in both cloud and an FN system was presented. Xuan *et al.* [31] considered a fog provider for task scheduling in a cloud-fog computing system, which can exploit the collaboration between its own FNs and the rented cloud nodes for executing the user's tasks. The major scope of this paper is to achieve the balance between the monetary costs and make span of cloud resources using a heuristics-based algorithm. Two parameters were defined for selecting the most suitable node to execute the task are the earliest start time (EST), and the earliest finish time (EFT). However, the scheduling algorithm needs to be robust against a deadline and budget constraints. Tejaswini *et al.* [32] proposed prioritized task scheduling in FC. This algorithm schedules many client requests based on their deadlines. Hence the proposed prioritized algorithm reduced the task execution time and also reduces the decreased cost due to prioritize tasks efficiently based on their levels and results of delay tolerance in higher overall throughput. But considering its energy efficiency, it takes higher energy processing single tasks. Thanh *et al.* [33] considered a fog based region and cloud (FBRC) where client requests are handled locally by multiple regions when extra resources are required. This paper resolved two major problems and presented a fog-based region architecture to provide computing resources by nearby nodes and instigating efficient scheduling algorithms to distribute the tasks among remote clouds and regions. However, region-based cloud infrastructure had fulfilled resource and sensitive latency requirements but did not consider the energy consumption problem.

Intharawijitr *et al.* [34] proposed 5G cellular network-enabled FC architecture to reduce the latency between the FNs to cloud data centers. With the support of the random policy, maximum available policy, and lowest policy, the target FN was selected. The simulation results obtained show a better performance due to its speedy available resource. Besides that, they found optimal value for the latency threshold in terms of the blocking probability. Oueis *et al.* [35] presented 5G future wireless networks for improving the user's quality of experience (QoE). For that, the authors proposed a small cell cluster with low complexity creation and resources management customizable algorithms for fog clustering. The simulation results showed that the proposed algorithm obtained high user satisfaction and a percentage of a minimum of 90% for up to 4 users per small cell, high latency gain, and moderate power consumption.

3. PROBLEM STATEMENT

In this section, we provide a formal and brief description of our problem considering task scheduling and data transmission by stating it into open issues. Ningning *et al.* [36] have proposed a dynamic graph partitioning theory to build the load balancing FC framework. Furthermore, they adopt Cloud Atomization technology to turn physical nodes in different levels into virtual machine nodes. The proposed algorithm reduces high power consumption and high latency, but it is not applicable for large scale cloud-FC infrastructure. Bitam *et al.* [37] have proposed the bees life algorithm (BLA) to address the problem of job scheduling in the FC environment. The proposed approach uses an optimized distribution approach to find an optimal tradeoff between allocated memory and CPU execution time required by FC services established by users of the mobile. The evaluation results of the empirical performance show the proposed approach outperforms in terms of allocated memory, and CPU execution time than the conventional optimization approaches such as genetic algorithm and particle swarm optimization. However, the performance of the overall proposed system is poor during end-to-end processing. Deng *et al.* [38] have proposed optimal workload allocation in a fog-cloud computing environment to reduce power and delay consumption. This framework provides a formal description of collaboration between the core (cloud) and the edge (fog) and formulated optimal workload allocations between cloud and fog using an approximate approach. Therefore, the proposed approach minimizes power consumption and task latency, but it leads to single node failure due to the centralized design of fog infrastructure.

We are introducing a new FC environment that solves the task scheduling problem using the WRR scheduling algorithm. The optimal FN is selected based on the number of tasks running currently in the system. WRR Scheduling algorithm assigns tasks to FNs based on their available processing capacity and remaining energy. Thus, the scheduling algorithm considers the pre-emptive tasks and tasks with no priority.

4. PROPOSED METHOD

Optimal Node selection is the central part of task scheduling, and it also enhances the overall time of response and reduces high energy consumption. This paper considers a cloud-fog environment. As illustrated in Figure 2, the FC architecture is composed of four layers: data source layer, FC layer, communication layer, and cloud layer. The proposed algorithm is employed in the fog layer, which initially checks if the computing resources are available in the FNs will meet the requirements of user tasks. When the resource is not enough or available for particular requests, then the task is transferred to the cloud layer. Also, the WRR scheduling algorithm processes all the tasks and serves them according to their available processing capability and remaining energy. The proposed model is represented below and illustrated in Figure 2.

a. The cloud layer is on the top, composed of cloud data centers.
b. In the next layer communication layer, it uses 5G networks for data communication and transmission. For routing, we used spanning-tree routing protocol (STP).
c. In the middle layer is the fog layer. It has many FN.
d. In the bottom layer is the data source layer where smart devices send tasks to its nearby FN.
e. Upon receiving the user tasks, the FN executed the following steps:
   - If the task is cannot be served by its constraints, the task is rejected
   - Otherwise, execute the task one to another FN according to its priority level, its available processing capacity, and remaining energy.

The pseudocode for the proposed algorithm is shown in Figure 3.

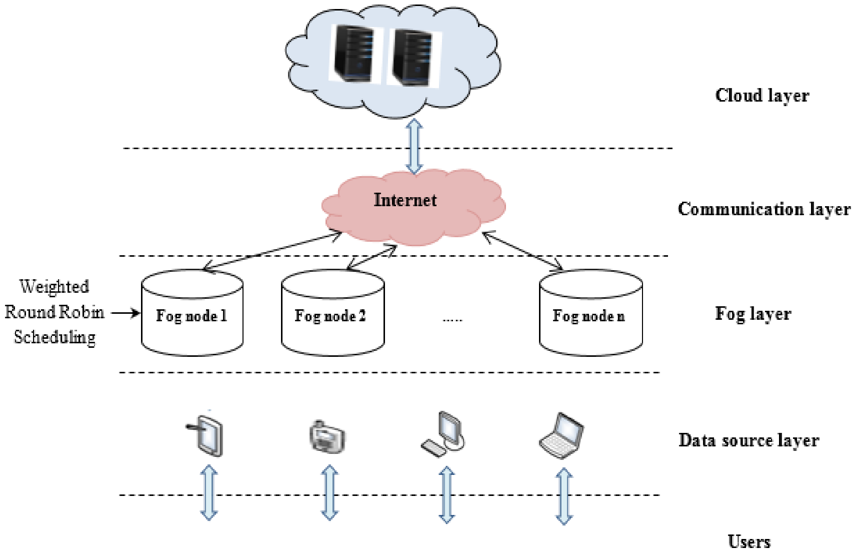
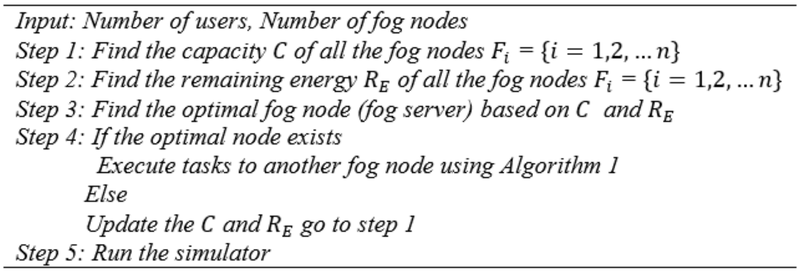


Figure 2. System architecture



```
Input: Number of users, Number of fog nodes
Step 1: Find the capacity C of all the fog nodes $F_i = \{i = 1,2, \dots n\}$
Step 2: Find the remaining energy $R_E$ of all the fog nodes $F_i = \{i = 1,2, \dots n\}$
Step 3: Find the optimal fog node (fog server) based on C and $R_E$
Step 4: If the optimal node exists
            Execute tasks to another fog node using Algorithm 1
        Else
        Update the C and $R_E$ go to step 1
Step 5: Run the simulator
```

Figure 3. The pseudocode for the proposed algorithm

4.1. WRR algorithm

In this paper, the task scheduling problem is overcome by the WRR Algorithm where each task was in turn executed within the cycle and consigns more time slice on some tasks that can finish in more than time. The WRR algorithm provides the application in FC which is a great enhancement for the overall operating efficiency and gives fairness for the proposed system.

Our proposed WRR algorithm allocates the number of user tasks within the rotation cycle (RC) based on the job-priority proportion than that of the system. The execution task number for a job in the cycle is having fewer weights and many residual tasks. Hence to get the currently running tasks number in the ith job cycle, we should search first for the job weight [i] using (1).

$$\frac{D(i)}{D} = \frac{\mu TS[i]*w[i]}{D} = \frac{P(i)}{\sum_{i=1}^{n} P(i)} \tag{1}$$

In (1) µTS[i] denotes the average size of the job task, D(i) denotes the data processing amount of this cycle, D is the amount of data processing that written in the systems, D(i) is the priority of job (i) and D(i) must be equal to the job’s priority proportion. The summation of all jobs based on its priority can be calculated as in (2).

$$w(i) = \frac{D*P(i)}{\mu TS(i)*\sum_{i=1}^{n} P(i)} \tag{2}$$

For every user job (i), the amount of data processing (D) is unknown in the whole system and set m to deal with the number of running tasks. Therefore, our proposed WRR solves the issues that response time is longer for small jobs to provide certain fairness.

4.2. Spanning tree routing protocol

Spanning tree protocol (STP) is primarily designed for data transmission of packets in a networking environment [39]. This protocol is highly capable of connecting the number of nodes to establish multiple paths for data transmission. It is also referred to as a distributed algorithm that assists network nodes to automatically adjust to link failures, particularly solves the single node failure problem. STP is enabled into the IEEE media access control standard 802.1D and serves as a basic element of local area network (LAN) architecture.

5. RESULTS AND DISCUSSION

In this section, we describe the experimental study in which conducted a set of simulation tests based on FC infrastructures. FNs are heterogeneous in terms of their storage capacity and processing capability. Table 1 shows the review for some related works, including contributions, advantages, and disadvantages of those works.

Table 1. Review of the related works presented

Methodology	Contributions	Advantages	Disadvantages
Intharawijitr *et al.* in [34]	Creating 5G fog based infrastructure	Reduce computing and communication latencies	Simulated on a partial fog computing system
Deng *et al.* in [38]	The use of integer nonlinear programming, Convex representation, The use of Hungarian method	Reduce power consumption and computation latency	Single node failure due to a centralized fog computing infrastructure
Ningning *et al.* in [36]	Graph partitioning representation, Minimum spanning tree, The use of multiple virtual machines	Reduce tasks run times	High load balancing complexity
Oueis *et al.* in [35]	Quality of Experience	Reduce power consumption and task latency	High complexity for large scale fog computing infrastructures
Bitam *et al.* in [37]	Bees life algorithm (BLA) for job scheduling problem	Reduce CPU execution time and allocated memory	Overall end-to-end processing performance leads to poor.

5.1. Simulation setup

This paper implemented with computer configuration is Pentium ® Dual Core CPU, E5700@3.00 GHz, RAM -2.00 GB, operating system for Ubuntu 14.04 (32-bit operating system, x64-based processor). To simulate the network, NS 3.26 (network simulator 3) and ifogsim simulator are used. Firstly, we declare the number of users and created fog devices by setting the parameters of FNs such as add node name, the rate per MIPS, bandwidth, a sensor with ifogsim, and power. The parameters setting for simulation are adopted as in Table 2.

Figure 4 shows the simulation environment for smart city applications including predictive maintenance, and health monitoring systems. Here introduced a new group of stringent requirements like latency (low latency) requirement since resources of computing can be on-demand requested concurrently by several devices at multiple locations and it is implemented using spanning tree protocol. Next, the FN can be used for executing the task to another FN to the cloud system using a WRR algorithm. Based on the results from the scheduling algorithm, the task is scheduled to FN and then to the cloud system for allocating resources. Figure 5 shows the optimal FN selection.

Table 2. Simulation parameters and values

Parameter	Value	Parameter	Value
Max. Cycles	100	Local Link Transmission Delay	10
Switch	100_SW	Global Link Transmission Delay	100
Xbar Delay	3	Injection Delay	1
Internal Speedup	2	Packet size	8
Arbiter Iteration	2	Flit size	8
Injection Queue Length	1000	SP Tree Configuration	2
Local Queue Length	32	Local Link Channel	3
Global Queue Length	256	Global channels	2
Out Queue Length	32	Probability	0.5
Ring Injection Bubble	1	Seed	10

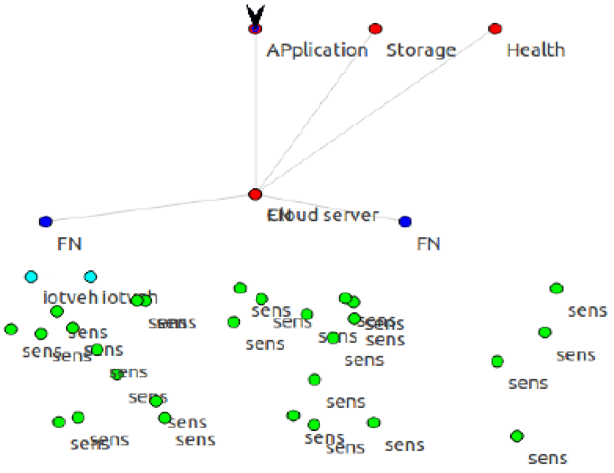


Figure 4. Smart city simulation environment

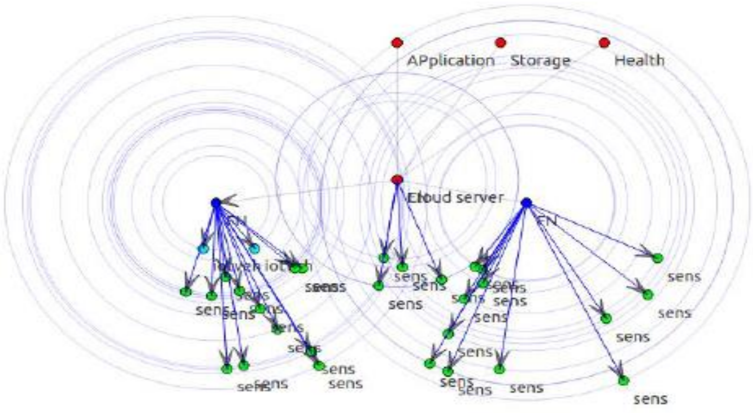


Figure 5. Optimal fn selection

5.2. Comparative analysis

In this section, we describe the comparative results for the proposed system with previous Fogflow approach [27].

5.2.1. Comparative analysis on throughput

Throughput is an important metric that is widely used in a networking environment to evaluate system performance. Our proposed system shows better performance than previous solutions for throughput. The comparative analysis of throughput is shown in Figure 6. From the graph, we observe that the existing system in [27] much less throughput than the proposed system. This is mostly because FogFlow broker never considers the available processing capacity and remaining energy for scheduling.

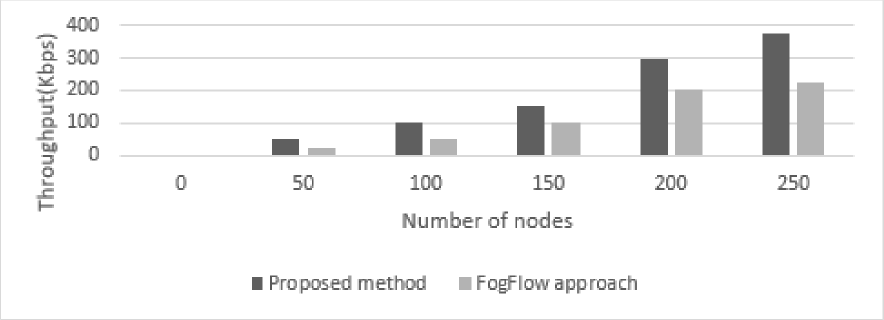


Figure 6. Comparative analysis of throughput

5.2.2. Comparative analysis of delay

Delay is another major performance metric that needs to be considered for the simulation environment. If the FNs are located at different locations, the latency (delay requirement) significantly increases. Figure 7 illustrates the comparative results for the proposed system as well as the existing system in [27] using delay. We observe that the proposed system takes less time for routing since we used STP for data collection and routing. Furthermore, we select the optimal node for executing the user tasks. Thus, our proposed system decreases the delay as the number of tasks running increases.

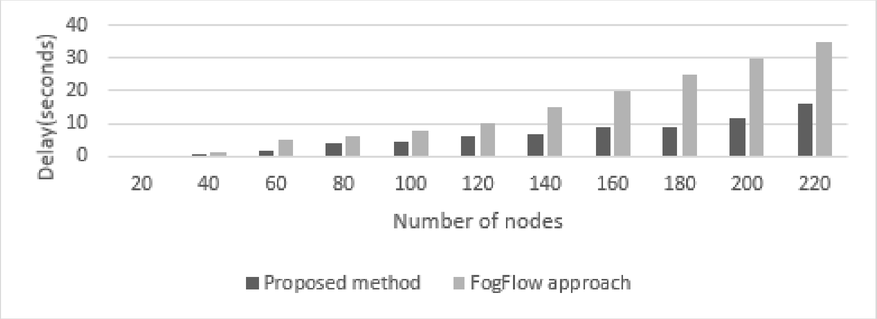


Figure 7. Comparative analysis of delay

5.2.3. Reduced complexity

Our proposed FC infrastructure obtained less computational complexity. This is mainly because we have used spanning tree routing protocol for data routing, WRR algorithm for task scheduling [40, 41], and fast data transmission using 5G networks. Based on the algorithms and techniques that we have used in this paper; we have obtained less complexity. Figure 8 shows the result of reduced complexity

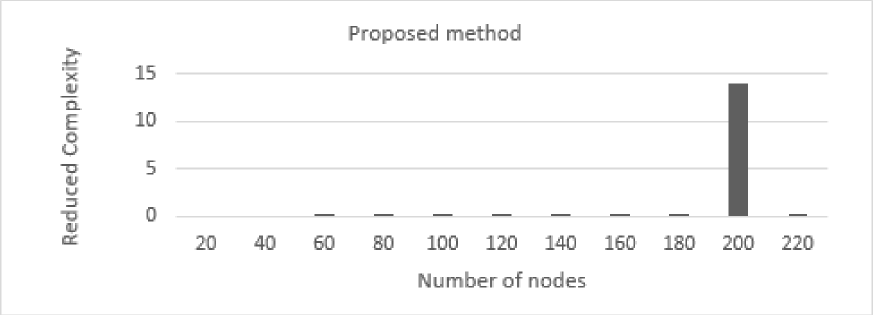


Figure 8. Results for reduced complexity

6. CONCLUSION AND FUTURE WORK

In the 21st century, the FC paradigm is expected to remain of interest for the researchers in industry and academia provide its incredible potential where services and computing resources are distributed in effective FNs reside at the cloud computing network edge. In this paper, we concentrated on the task scheduling problem in the environment of FC to assure the effective task execution according to the available processing capacity and remaining energy. To address the open issue i.e., scheduling of tasks in the environment of FC, a new scheduling algorithm called WRR scheduling has been proposed. To deal with the task scheduling problem in FC, two performance evaluation metrics were used in this work, namely, available processing capacity, and remaining energy required by all tasks expected to be performed in the FC environment. For efficiency evaluation, we executed a set of simulation evaluations on the implementation of scheduling and compared obtained results against the results of the approaches of conventional task scheduling in terms of response time and latency. In the future, we have planned to implement a dynamic task scheduling algorithm that needs to consider the arrival of new requests while the other request is being executed for the FC environment.

REFERENCES

[1] Zhang C., “Design and application of fog computing and Internet of Things service platform for smart city,” *Future Generation Computer Systems*, vol. 112, pp. 630-640, 2020.

[2] Puliafito C., Mingozzi E., Longo F., Puliafito A. and Rana O., “Fog computing for the internet of things: A Survey,” *ACM Transactions on Internet Technology (TOIT)*, vol. 18, no. 2, pp. 1-41, 2019.

[3] Naeem R.Z., Bashir S., Amjad M.F., Abbas H., and Afzal H., “Fog computing in internet of things: Practical applications and future directions,” *Peer-to-Peer Networking and Applications*, vol. 12, no. 5, pp. 1236-1262, 2019.

[4] Lavanya R., “Fog computing and its role in the Internet of Things. Advancing Consumer-Centric Fog Computing Architectures,” *IGI Global*, pp. 63-71, 2019.

[5] Saravanan K., Julie E.G., and Robinson Y.H., “Smart cities & IoT: evolution of applications, architectures & technologies, present scenarios & future dream,” *Intelligent Systems Reference Library*, pp. 135-151, 2019.

[6] Al-Smadi A.M., Alsmadi M.K., Baareh A., Almarashdeh I., Abouelmagd H., and Ahmed O.S.S., “Emergent situations for smart cities: a survey,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 6, pp. 4777-4787, 2019.

[7] Giordano A., Spezzano G. and Vinci A., “Smart agents and fog computing for smart city applications,” In *International Conference on Smart Cities*, vol. 9704, pp. 137-146, 2016.

[8] Kaur M.J. and Maheshwari P., “Building smart cities applications using IoT and cloud-based architectures,” *2016 International Conference on Industrial Informatics and Computer Systems (CIICS)*, Sharjah, 2016, pp. 1-5.

[9] Hammi B., Khatoun R., Zeadally S., Fayad A. and Khoukhi L., “IoT technologies for smart cities,” *IET Networks*, vol. 7, no. 1, pp. 1-13, 2018.

[10] Suzuki L.R., “Smart cities IoT: Enablers and technology road map,” *Smart City Networks*, pp. 167-190, 2017.

[11] Kim Tai-hoon, Ramos C. and Mohammed S., “Smart city and IoT,” *Future Generation Computer Systems*, vol. 76, pp. 159-162, 2017.

[12] Oteafy S.M., and Hassanein H.S., “IoT in the fog: A roadmap for data-centric IoT development,” *IEEE Communications Magazine*, vol. 56, no. 3, pp. 157-163, 2018.

[13] Shahidehpour M., Li Z., and Ganji M., “Smart cities for a sustainable urbanization: Illuminating the need for establishing smart urban infrastructures,” *IEEE Electrification Magazine*, vol. 6, no. 2, pp. 16-33, 2018.

[14] Wakefield J., “Tomorrow's cities: Do you want to live in a smart city,” *BBC News*, 2013.

[15] Kientopf K., Raza S., Lansing S., and Güneş M., “Service management platform to support service migrations for IoT smart city applications,” *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Montreal, QC, 2017, pp. 1-5.

[16] Bibri S.E., “On the sustainability of smart and smarter cities in the era of big data: an interdisciplinary and transdisciplinary literature review,” *Journal of Big Data*, vol. 6, no. 1, p. 25, 2019.

[17] Rahbari D., and Nickray M., “Scheduling of fog networks with optimized knapsack by symbiotic organisms search,” *2017 21st Conference of Open Innovations Association (FRUCT)*, Helsinki, 2017, pp. 278-283.

[18] F. Maroto, “What is the role of Fog Computing in the IOT reference model,” *bigdata-madesimple*, 2016. [Online]. Available: https://bigdata-madesimple.com/what-is-the-role-of-fog-computing-in-the-iot-reference-model/.

[19] Perera C., Qin Y., Estrella J.C., Reiff-Marganiec S., and Vasilakos A.V., “Fog computing for sustainable smart cities: A survey,” *ACM Computing Surveys*, vol. 50, no. 32, pp. 1-43, 2017.

[20] Aleisa M.A., Abuhussein A. and Sheldon F.T., “Access Control in Fog Computing: Challenges and Research Agenda,” *IEEE Access*, vol. 8, pp. 83986-83999, 2020.

[21] Baccarelli E., Naranjo P.G.V., Scarpiniti M., Shojafar M. and Abawajy J.H., “Fog of everything: Energy-efficient networked computing architectures, research challenges, and a case study,” *IEEE Access*, vol. 5, pp. 9882-9910, 2017.

[22] Zahmatkesh H. and Al-Turjman F., “Fog computing for sustainable smart cities in the IoT era: Caching techniques and enabling technologies-an overview,” *Sustainable Cities and Society*, vol. 59, p. 102139, 2020.

[23] Hu F., Qiu M., Li J., Grant T., Taylor D., McCaleb S., Butler L., and Hamner R., “A review on cloud computing: Design challenges in architecture and security,” *Journal of computing and information technology*, vol. 19, no. 1, pp. 25-55, 2011.

[24] Chiang M. and Zhang T., “Fog and IoT: An overview of research opportunities,” *IEEE Internet of Things journal*, vol. 3, no. 6, pp. 854-864, 2016.

[25] Bruneo D., et al., “Stack4Things as a fog computing platform for Smart City applications,” *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS),* San Francisco, CA, 2016, pp. 848-853.

[26] Santos J., Wauters T., Volckaert B. and De Turck F., “Fog computing: Enabling the management and orchestration of smart city applications in 5G networks,” *Entropy*, vol. 20, no. 1, pp. 1-4, 2018.

[27] Cheng B., Solmaz G., Cirillo F., Kovacs E., Terasawa K. and Kitazawa A., “Fogflow: Easy programming of IoT services over cloud and edges for smart cities,” *IEEE Internet of Things journal*, vol. 5, no. 2, pp. 696-707, 2017.

[28] Mahmud R, Kotagiri R and Buyya R., “Fog computing: A taxonomy, survey and future directions,” *Internet of everything*, pp. 103-130, 2018.

[29] Mohamed N., Al-Jaroodi J., Jawhar I., Lazarova-Molnar S. and Mahmoud S., “SmartCityWare: A service-oriented middleware for cloud and fog enabled smart city services,” *IEEE Access*, vol. 5, pp. 17576-17588, 2017.

[30] Sinaeepourfard A., Garcia J., Masip-Bruin X. and Marin-Tordera E., “A novel architecture for efficient fog to cloud data management in smart cities,” *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS),* Atlanta, GA, 2017, pp. 2622-2623.

[31] Pham X.-Q., and Huh E.-N., “Towards task scheduling in a cloud-fog computing system,” *2016 18th Asia-Pacific Network Operations and Management Symposium (APNOMS),* Kanazawa, 2016, pp. 1-4.

[32] Choudhari T., Moh M. and Moh T.-S., “Prioritized task scheduling in fog computing,” In Proceedings of the *ACMSE 2018 Conference*, no. 22, 2018, pp. 1-8.

[33] Hoang D. and Dang T. D., “FBRC: Optimization of task scheduling in fog-based region and cloud*,” 2017 IEEE Trustcom/BigDataSE/ICESS*, Sydney, NSW, 2017, pp. 1109-1114.

[34] Intharawijitr K., Iida K. and Koga H., “Analysis of fog model considering computing and communication latency in 5G cellular networks,” *2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops),* Sydney, NSW, 2016, pp. 1-4.

[35] Oueis J., Strinati E. C. and Barbarossa S., “The fog balancing: Load distribution for small cell cloud computing,” *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*, Glasgow, 2015, pp. 1-6.

[36] Ningning S., Chao G., Xingshuo A. and Qiang Z., “Fog computing dynamic load balancing mechanism based on graph repartitioning,” *China Communications*, vol. 13, no. 3, pp. 156-164, 2016.

[37] Bitam S., Zeadally S. and Mellouk A., “Fog computing job scheduling optimization based on bees swarm,” *Enterprise Information Systems*, vol. 12, no. 4, pp. 373-397, 2018.

[38] Deng R., Lu R., Lai C., Luan T. H. and Liang H., “Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption,” *IEEE Internet of Things journal*, vol. 3, no. 6, pp. 1171-1181, 2016.

[39] Nadeem M. A. and Karamat T., “A survey of cloud network overlay protocols,” *2016 Sixth International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*, Konya, 2016, pp. 177-182.

[40] Zhang Y. and Harrison P. G., “Performance of a priority-weighted round robin mechanism for differentiated service networks,” *2007 16th International Conference on Computer Communications and Networks*, Honolulu, HI, 2007, pp. 1198-1203.

[41] Jung S., Kwak J. and Byeon O., “Performance analysis of queue scheduling mechanisms for ef phb and af phb in diffserv networks,” *5th IEEE International Conference on High Speed Networks and Multimedia Communication (Cat. No.02EX612)*, Jeju Island, South Korea, 2002, pp. 101-104.