ISSN 2347 - 3983
Volume 9. No. 7, July 2021
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter32972021.pdf
https://doi.org/10.30534/ijeter/2021/32972021

Dynamic Content Enabled Microservice for Business Applications in Distributed Cloudlet Cloud Network

Mujeeb ur Rehman Shaikh[1], Anwar Ali Sathio[2], Abdullah Lakhan[3],Ali Orangzeb Panhwar[4], M.Afzal Sahito[5],Asadullah Kehar[6]
[1, 2,3,4,5] Lab of Information Security and AI, Department of Computer Science and Information Technology,Benazir Bhutto Shaheed University lyari,Sindh, Pakistan.
mujeebshaikh137@gmail.com , anwar.sathio@bbsul.edu.pk, abdullahrazalakhan@gmail.com, Orangzebpanhwar@gmail.com , Afzalsahito87@gmail.com ,
[6]Department of Computer Science Shah Abdul Latif University Khairpur, asad.kehar@salu.edu.pk

ABSTRACT

This study introduces Mob-Cloud, a mobility aware adaptive offloading system that incorporates a mobile device as a thick client, ad-hoc networking, cloudlet DC, and remote cloud to improve the performance and availability of microservices services. These cloudlet cloud has emerged as a popular model for bringing the benefits of cloud computing to the proximity of mobile devices. the microservices preliminary goal is to improve service availability as well as performance and mobility features. The impact of dynamic changes in mobile content (e.g., network status, bandwidth, latency, and location) on the task offloading model is observed by proposing a mobility aware adaptive task offloading algorithm aware microservices, which makes a task offloading decision at runtime on selecting optimal wireless network channels and suitable offloading resources. The decision problem, which is well-known as an NP-hard issue, is the subject of this work. However, for the entire proposed microservices system has the following phases: I adaptive offloading decision based on real-time information, (ii) workflow task scheduling phase, (iii) mobility model phase to motivate end-users to invoke cloud services seamlessly while roaming, and (iv) faulttolerant phase to deal with failure (either network or node). We conduct real-world experiments on the built instruments to assess the online algorithm's overall performance. Compared to baseline task offloading solutions, the evaluation findings show that online algorithm incorporates dynamic adjustments on offloading decision during run-time and achieves a massive reduction in overall response time with better service availability.

Key words : Microservices, Cloudlet, Cloud, Offloading, Mobility module.

1. INTRODUCTION

These days, the usage business workflow applications are growing progressively [1]. The applications are E-Ticket, E-Purchasing, E-Searching, Booking flights and so are more practical [2]. Therefore, all business applications are more cost-efficient and dynamic to these applications have quality of service requirements (QoS) to any system [3]. The cloud computing is an emerging paradigm which offers different services to run these applications with the economic cost and average cost range services for these applications [4]. The cloud computing has the different computing model and resource provisioning such as infrastructure as a service, platform as a service and software as a service [5]. The cloud services for the business applications are more effective in terms of cost and resource availability [6]. However, due to placement of multiple hops away from the internet based on service of cloud computing, it incurred with the long latency problem for the business applications [7]. In order to reduce end to end delay and latency, efficient aware cloudlet paradigm proposed in different studies [8,9,19,11]. The cloudlet is the new paradigm which brings remote cloud services at the edge of the mobile network. The bandwidth utilization, cost and services can be minimized by the cloudlet [12].

There are many nodes are connected for the distributed business applications. For instance, mobile node, network node and cloud node to perform any single process of business applications [13]. However, there are many challenges exist in the cloudlet cloud network for business applications [14]. (1) The dynamic content of network could be changed during the mobile of applications, therefore, static offloading cannot meet the requirements of dynamic workflow business applications [15]. (2) Existing service oriented architecture (SOA) offers cloud and cloudlet centralized services which has a failure issue in the network [16]. (3) The static and design time profiling of applications can not adopt the dynamic changes of applications in the current content aware cloudlet cloud system. Many studies have already presented task offloading solutions in the CLOUDLET CLOUD environment for mobile workflow applications in the literature [17,18,19,20,21,22,23,24,25]. The earlier hypothesis, on the other hand, seems highly implausible. A wireless network context varies intermittently owing to traffic at different times, as it does in real practise. Cloud service fluctuations, on the other hand, are common in different time zones. With recent advancements in wireless technology, a mobile device now has access to a variety of wireless channels, including WiFi, cellular networks, and Bluetooth. In terms of speed and strength, each connection performs differently. We have diverse communication methods and cloud paradigms for mobile workflow applications, therefore mobility and fault aware task offloading in a dynamic context is a difficult problem to address. In an adaptable ecosystem, the following questions have to be answered. This study suggests a novel content-aware microservices offloading system for workflow applications, which can solve the dynamic offloading of the survey.

The paper makes the following contribution in the study.

1. The study devises a novel content awre services that is called Microservice for the business workflow applications. The proposed system is a dynamic and conent aware in the system.
2. The sysem can handle any kind of failure in the system.
3. The node failure and network failure can handle in the system.
4. The study devises the online offloading aware schemes for business applications in the system.

The rest of the paper is organized as the following. Sections 2 discusses the related work of the studies. Section 3 is the proposed schemes. Section is the conclusion and future work.

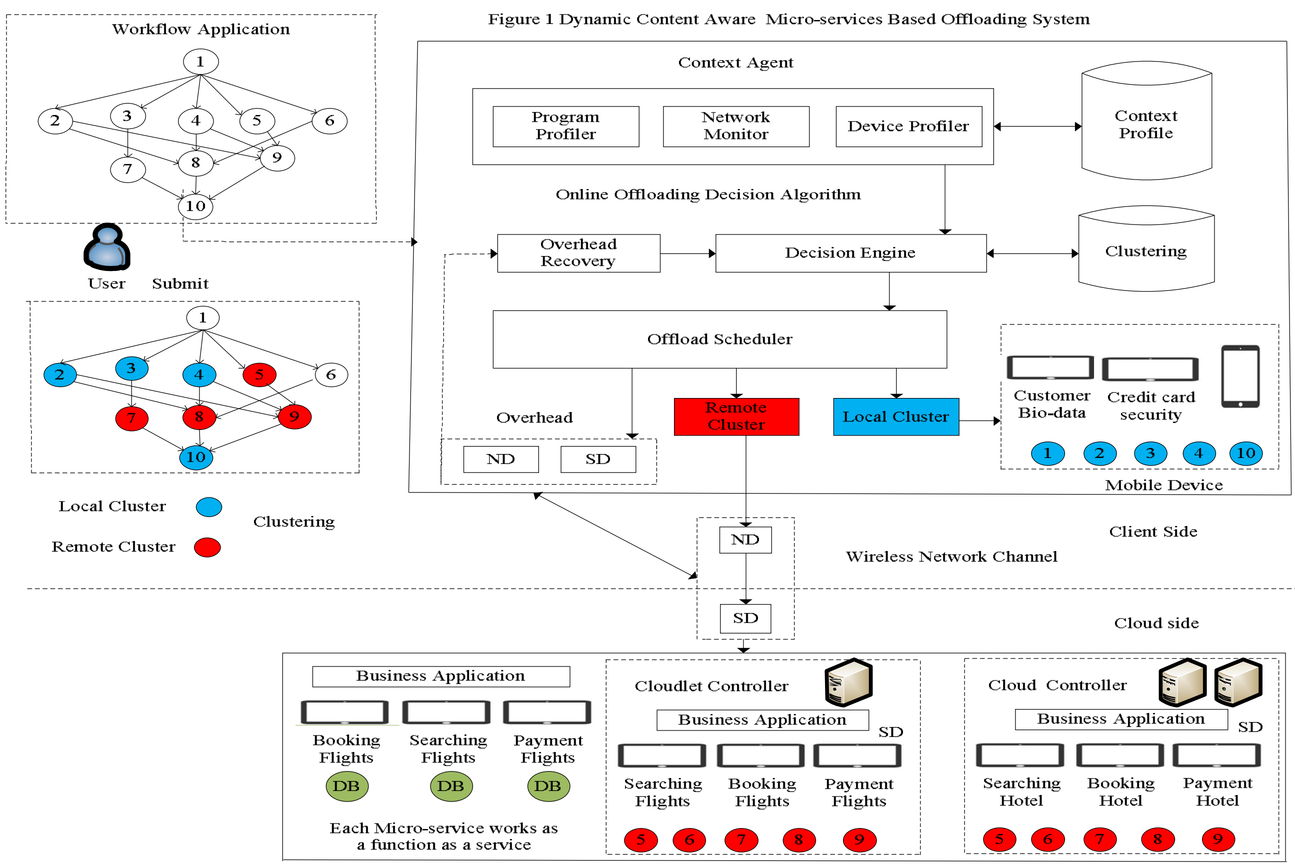

Figure 1 Proposed System

2. RELATED WORK

Many extensive studies on task offloading in the dynamic cloudlet cloud environment have already been published in the literature. However, in terms of computer power, the latest technologies IoT (Internet of Things) devices differ. Nonetheless, the proposed microservice cloud in this study is not limited to mobile devices; rather, it focuses on mobile devices as IoT devices that offload compute-intensive tasks to the cloud for processing. Mobility management and fault-tolerant task offloading strategies, on the other hand, would meet the service needs for IoT devices. The following papers have presented several task offloading algorithms in CLOUDLET CLOUD environments: Chun and colleagues proposed the CloneCloud offloading architecture in [5]. The goal was to extend the battery life of mobile phones and increase application performance.

The application partitioning for conducting the computational offloading is done at the thread level granularity [6]. Proposes the Think-Air energy-efficient computational offloading paradigm. The major goal was to reduce mobile device energy usage while also extending battery life. The proposed framework performs computational offloading at the code level (binaries). Cuckoo, MAUI, and JADE proposed their energy-efficient frameworks for computational offloading in [7–9], regardless of whether offloading occurs during runtime or not. The fundamental goal of the run time offloading decision was to extend the battery life of mobile devices while also easing the burden on application developers. Delay-sensitive applications must run in a short amount of time; yet, because modern cloud services are available over long distance WAN, there may be a lengthier end-to-end delay. Satyanarayanan et al. suggest a Cloudlet framework to address the same challenge in [10]. The main goal is to bring cloud computing capabilities closer to mobile users in order to reduce total latency for time-sensitive applications. Many of the previous studies, on the other hand, concentrated on task offloading mechanisms without taking into account mobility aspects in an adaptable environment.

In order to enable delay-sensitive applications, an offloading technique based on mixed fog and cloud architecture is presented [12]. In their different publications [13,14,15], the primary purpose was to examine how to execute application segmentation in a dynamic environment in MCC architecture to invoke cloud services. These papers[16, 17, 19] studied latency aware optimal workload assignment to cloud computing, but [19] considered a heterogeneous mobile cloud environment (ad-hoc, edge cloud, and remote) to make offloading decisions. [20, 21] advocated a decentralised mobile cloud environment to improve task offloading resource availability. These work were centred on minimising delay for real-time applications under the MCC paradigm.

In today's world, all IoT device is mobile; nevertheless, in a dynamic context, mobility and fault-aware adaptive job offloading would be more suited.

Mobility aware adaptive task offloading with a fault tolerant method has not been researched to the best of our knowledge. We suggested the MATOA algorithm framework to enhance the functionality of the task offloading strategy, which deals with mobile application performance while travelling and invoking appropriate services in the shadow of fault-tolerant awareness.

3. PROPOSED SYSTEM

In heterogeneous cloud networks, mobility and fault-aware adaptive job offloading are a difficult problem to solve.

As a result, managing application duties in a dynamic environment is a difficult process. Therefore, we proposed a mob-cloud design that is comparable to existing systems in [22]. The mob-cloud architecture has two primary components: a client-side component and a cloud-side component, as shown in Fig. 1. On the client-side, the Content Agent module, Decision Engine module, Workflow Scheduler module, and Failure Manager module are the four primary modules. At runtime, the Content Agent module collects real-time data on various characteristics, including the application programme, current network bandwidth, latency, roundtrip time, and cloud resource estimation [22].

On the other hand, it aids the Decision Engine module make adaptive task offloading decisions based on real-time data from different parameters. The decision engine module then splits the resilient workflow application into two clusters of tasks: a local cluster (e.g., lightweight tasks) and a remote cluster (e.g., heavyweight tasks) (e.g., compute-intensive tasks). The Workflow Scheduler module is in charge of task scheduling and has two basic functions: first, it schedules the local cluster of tasks on heterogeneous mobile cores; and second, it schedules the local cluster of tasks on the heterogeneous mobile cores. second, allocate a suitable accessible efficient wireless network channel to the cloud server for the remote cluster of processes. It's worth noting that the Workflow Scheduler can only schedule one task at a time and can do so on a mobile device or over a wireless channel network. It can't schedule tasks on the cloud server, but it can anticipate the task process time. During the runtime of an application, the Failure Manager module plays a more vital function, identifying two preliminary sub failure modules, such as a network failure module (NF) and a node or resource failure module (RF). Because network contents fluctuate owing to a variety of factors, such as weak signals, high communication delay, and lesser bandwidth, overall performance may suffer. As a result of the changing network context, the transmission time for offloading and downloading data for running applications may exceed the deadline. As a result, numerous application tasks may fail owing to either a weak network or network unavailability.

The NF module adaptively notifies the failure manager of unsuccessful tasks and displays a list of alternative wireless networks. The RF module, on the other hand, keeps track of a remote set of task execution processes on assigned resources. When it identifies a failure owing to node capacity or unavailability, it gathers a tuple of data (e.g., task ID, location ID, site of failure, and result obtained) and sends it back to the failure manager for rescheduling. It's worth noting that failure can only be recognised after scheduling. Partitioning of applications. In most cases, a mobile workflow application comprises composite dependent components (e.g., tasks). Some of them, though, The application's compute-intensive components will transfer their data and operations to the cloud Using the partitioning approach There are numerous approaches.for partitioning applications, such as programmerdefine an approach, profile-based linear programming, Last but not least, simulation-based, graph-based detail on the inference algorithm and the partitioning methods [22] can be displayed. In this work, we will solely discuss because of the importance of profile-based partitioning Considered an issue, mob-cloud offers a lot of settings to play with. do application partitioning, i.e. the cost of task execution Bandwidth, CPU speed, network type, and cloud service are all factors to consider. As shown, there is availability and a round-trip time. The To make an offloading decision, offer the microservice algorithm. based on the collection of real-time information given by several profilers, as well as programme, Individually, network, and resource profiler Characterization of the task After the partitioning procedure, the application can be divided into many kinds. fine-grained objects, modules, classes, bundles, and threads are all examples of fine-grained objects. components and functionalities of the system.

4. CONCLUSION

The dynamic content-aware task offloading problem in diverse mobile cloud settings is investigated in this paper. Failure aware task offloading in an adaptive environment, to the best of our knowledge, is useful for real-world applications. We added additional components to the current MCC architecture to allow user mobility, fault tolerance, and task scheduling for a workflow application.

Under task priority limitation and hard constraint deadline, we suggested a novel Mobility Aware Task Offloading Algorithm that minimises the workflow application's average response time and delays the sensitive application. The online offloading algorithm has three phases:

1. It implements a multi-criteria task offloading mechanism.
2. It implements a modified DHEFT heuristic to schedule local and remote cluster tasks on appropriate resources.
3. It implements a failure mechanism to support the end user's mobility and interactivity in a dynamic environment.

Because existing modified mob-cloud provides cloud services as infrastructure as a service packaged in a virtual machine box, we will design a complex task offloading algorithm and mob-cloud architecture for future work. On the other hand, virtual machine-based services are heavyweight and require upfront pre-allocation of cloud resources (e.g., RAM, storage, libraries, host OS). Yet, requested tasks of an application may only use a tiny amount of resources, resulting in resource waste. Most IoT devices use microservices to complete their jobs, however single code virtual machine services are difficult to recover or re-deploy if they fail for any reason. In the future, we want to develop a mob-cloud based on container microservices to accommodate IoT device data better and increase the cloud server's resource usage. With advanced wireless technology and cloud computing advancement, mobility as a service, function as a service, and offloading as a service will become an effective architecture for smart devices. Because it is difficult to deploy complex workflow applications into smart devices directly, we will extend mob-cloud and add fog nodes in the future to preprocess IoT on edge nodes before offloading.

REFERENCES

1. Waseem, m., lakhan, a., & jamali, i. A. (2016). Data security of mobile cloud computing on cloud server. Open access library journal, 3(4), 1-11.
2. Lakhan, a., & li, x. (2020). Transient fault aware application partitioning computational offloading algorithm in microservices based mobile cloudlet networks. Computing, 102(1), 105-139.
3. Lakhan, a., ahmad, m., bilal, m., jolfaei, a., & mehmood, r. M. (2021). Mobility aware blockchain enabled offloading and scheduling in vehicular fog cloud computing. Ieee transactions on intelligent transportation systems.
4. Lakhan, a., & xiaoping, l. (2018, november). Energy aware dynamic workflow application partitioning and task scheduling in heterogeneous mobile cloud network. In 2018 international conference on cloud computing, big data and blockchain (iccbb) (pp. 1-8). Ieee.
5. Lakhan, a., & li, x. (2019). Mobility and fault aware adaptive task offloading in heterogeneous mobile cloud environments. Eai endorsed transactions on mobile communications and applications, 5(16).
6. Mahesar, a. R., lakhan, a., sajnani, d. K., & jamali, i. A. (2018). Hybrid delay optimization and workload assignment in mobile edge cloud networks. Open access library journal, 5(9), 1-12.
7. Mahesar, a. R., lakhan, a., sajnani, d. K., & jamali, i. A. (2018). Hybrid delay optimization and workload assignment in mobile edge cloud networks. Open access library journal, 5(9), 1-12.
8. Sajnani, d. K., mahesar, a. R., lakhan, a., & jamali, i. A. (2018). Latency aware and service delay with task scheduling in mobile edge computing. Communications and network, 10(04), 127.
9. Jamali, i. A., lakhan, a., mahesar, a. R., & sajnani, d. K. (2018). Energy aware task assignment with cost optimization in mobile cloud computing. International journal of communications, network and system sciences, 11(08), 175.
10. Jamali, i. A., lakhan, a., kumar, d., & mahessar, a. R. (2018). Energy efficient task assignment algorithm framework in mo-bile cloud computing. Gsj, 6(8), 171.
11. Soomro, a. A. (2015). The secure data storage in mobile cloud computing. Journal of information & communication technology (jict), 9(2), 9.
12. Hussain, m., wei, l. F., lakhan, a., wali, s., ali, s., & hussain, a. (2021). Energy and performance-efficient task scheduling in heterogeneous virtualized cloud computing. Sustainable computing: informatics and systems, 30, 100517.\
13. Lakhan, a., mohammed, m. A., rashid, a. N., kadry, s., panityakul, t., abdulkareem, k. H., & thinnukool, o. (2021). Smart-contract aware ethereum and client-fog-cloud healthcare system. Sensors, 21(12), 4093.
14. Khoso, f. H., lakhan, a., arain, a. A., soomro, m. A., nizamani, s. Z., & kanwar, k. (2021). A microservice-based system for industrial internet of things in fog-cloud assisted network. Engineering, technology & applied science research, 11(2), 7029-7032.

15. Khoso, f. H., arain, a. A., lakhan, a., kehar, a., & nizamani, s. Z. (2021). Proposing a novel iot framework by identifying security and privacy issues in fog cloud services network. Int. J, 9, 592-596.
16. Lakhan, a., memon, m. S., elhoseny, m., mohammed, m. A., qabulio, m., & abdel-basset, m. (2021). Cost-efficient mobility offloading and task scheduling for microservices iovt applications in container-based fog cloud network. Cluster computing, 1-23.
17. Xiaoping, l. (2018, december). Dynamic partitioning and task scheduling for complex workflow healthcare application in mobile edge cloud architecture. In 2018 ieee 4th international conference on computer and communications (iccc) (pp. 2532-2536). Ieee.
18. Lakhan, elhoseny, m., memon, m. S., & mohammed, m. A. (2021). Deep neural network-based application partitioning and scheduling for hospitals and medical enterprises using iot assisted mobile fog cloud. Enterprise information systems, 1-23.
19. Lakhan, a., & li, x. (2019, december). Content aware task scheduling framework for mobile workflow applications in heterogeneous mobile-edge-cloud paradigms: catsa framework. In 2019 ieee intl conf on parallel & distributed processing with applications, big data & cloud computing, sustainable computing & communications, social computing & networking (ispa/bdcloud/socialcom/sustaincom) (pp. 242-249). Ieee.
20. Lakhan, a., & xiaoping, l. (2018, november). Energy aware dynamic workflow application partitioning and task scheduling in heterogeneous mobile cloud network. In 2018 international conference on cloud computing, big data and blockchain (iccbb) (pp. 1-8). Ieee.
21. Lakhan, a., sajnani, d. K., tahir, m., aamir, m., & lodhi, r. (2018, december). Delay sensitive application partitioning and task scheduling in mobile edge cloud prototyping. In international conference on 5g for ubiquitous connectivity (pp. 59-80). Springer, cham.
22. Sun, y., he, q., qi, l., rafique, w., & dou, w. (2020, october). Dpoda: differential privacy-based online double auction for pervasive edge computing resource allocation. In proceedings of the 2nd acm international symposium on blockchain and secure critical infrastructure (pp. 130-141).
23. Sathio, A. A. (2018). A Study on the Conceptual Framework of Data warehousing in Health Sector in Pakistan A Case Study of a Hospital System and Disease (Hepatitis C). International Journal of Computer (IJC), 29(1), 59-81.
24. Ahmed Shaikh, Z., Ali Sathio, A., Ali Laghari, A., Ahmed Memon, M., & Hussain Mirani, I. (2019). Study of the Role of New Technologies in Pharmaceutical Industry. Journal of Pharmaceutical Research International, 31(6), 1-11. https://doi.org/10.9734/jpri/2019/v31i630313
25. Shaikh, Z. A., Wagan, A. A., Laghari, A. A., Ali, K., Memon, M. A., & Sathio, A. A. (2019). The role of software configuration management and capability maturity model in system quality. IJCSNS, 19(11), 114.