IJIRMET ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 9, Issue 3, April 2024

Traffic-Aware Dynamic Container Migration for Real-Time Support in Mobile Edge Clouds

[1] Dr. E.Punarselvam ,M.E., Ph.D., [2]B. Ajaykumar., [3]S. Kishorekumar., [4]G. Nivethan
[1] Professor, Department of Information Technology, Muthayammal Engineering College (Autonomous), Rasipuram - 637 408, Tamil Nadu, India
[2] [3] [4] Student, Department of Information Technology, Muthayammal Engineering College (Autonomous), Rasipuram - 637 408, Tamil Nadu, India

Abstract: ***Edge clouds face challenges of resource assignment and load balancing due to variability of user location (mobility), server load and network state. Dynamic resource migration techniques are considered necessary to achieve load balance, fault tolerance and system maintenance objectives. Container migration is emerging as a potential solution that enables dynamic resource migration in virtualized networks and mobile edge cloud (MEC) systems. This paper proposes a traffic aware container migration approach and validates it with an end-to-end system implementation using a pure container hypervisor called LXD (Linux Container Hypervisor). The container migration model is then evaluated for real-time applications such as license plate recognition running in a mobile edge cloud scenario based on city-scale mobility traces from taxicabs in San Francisco. The system evaluation considers key metrics associated with application quality-of-experience (QoE) and network efficiency such as the average system response time and the migration cost for different combinations of load, compute resources, interedge cloud bandwidth, network and user latency. A specific compute resource and network-aware distributed resource migration algorithm called ‖ShareOn‖ is proposed and compared with alternative techniques using the San Francisco MEC model.***

I. INTRODUCTION

Heterogeneous edge cloud refers to a computing infrastructure that combines various types of computing resources, including both processing power and storage capacity, located at the edge of a network. The term "edge" in computing generally refers to the point where data is generated, processed, and consumed, which is closer to the source of data rather than in a centralized data center.

In a heterogeneous edge cloud environment, the computing resources are diverse and may include different types of devices, servers, and platforms. This diversity allows for a more flexible and distributed computing infrastructure, catering to the specific requirements of different applications and services.

We propose Shareon, a traffic-aware container migration algorithm using LXD and CRIU (Checkpoint Restore in Userspace). An end-to-end migration framework running real-time application has been deployed to analyze the impact of resource allocation, latency, bandwidth, size and migration time. A simulation model is also set up in which the containers are hosted in an edge cloud network running an automated license plate recognition application. Real traces from taxicabs in San. Scalability of the system with respect to increasing traffic load is investigated using the above mentioned city-scale MEC model.

MEC AND CONTAINER MIGRATION

The architecture of MEC is based on three layers of computing, the first at the mobile client, the second at a local network attached edge cloud, and the third at a centralized data center/cloud in the core of the network. As shown in Fig. 1, this architecture offers the advantage of low latency response to real-time applications which cannot tolerate a typical edgeto-core round trip delay that typically exceeds ~100 ms. With the help of flexible resource provisioning and sharing among neighboring edge cloud nodes, MEC can meet the unpredictable traffic demands and quickly scale the network due to its multitenancy feature. Further, in order to reduce the application latency and to provide the required user QoE, service requests are handled by the resource virtualized environment. Container-based virtualization is finding increasing adoption in MEC systems to realize slice isolation and fine resource control. Resource isolation (especially, memory) across components of different applications is necessary for the integrity of individual applications.

II. PROPOSED SYSTEM

A compute and network-aware lightweight resource sharing framework with dynamic container migration, Share On, is proposed. The migration framework is validated using a set of heterogeneous edge cloud nodes distributed in San Francisco city, serving mobile taxicab users across that region. The end-to-end system is implemented using a container hypervisor called LXD (Linux Container Hypervisor) executing a real-time application to detect license number plates in automobiles. The system is evaluated based on key metrics associated with application quality-of-service (QoS) and network efficiency such as the average system response time and the migration cost for different combinations of load, compute resources, inter-edge cloud bandwidth, network and user latency.

ADVANTAGES

- Portability.
- Efficiency.
- Agility.
- Faster delivery.
- Improved security.
- Faster app startup.
- Easier management.
- Flexibility

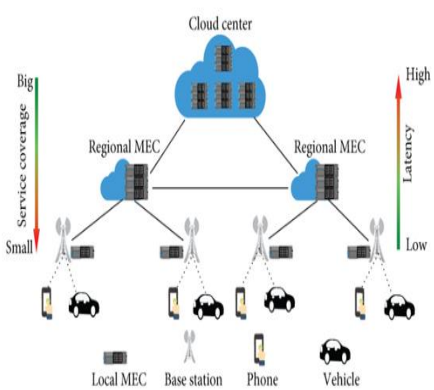
IMPLEMENTATION

The end-to-end system is implemented using a container hypervisor called LXD (Linux Container Hypervisor) executing a real-time application to detect license number plates in automobiles. The system is evaluated based on key metrics associated with application quality-of-service (QoS) and network efficiency such as the average system response time and the migration cost for different combinations of load, compute resources, inter-edge cloud bandwidth, network and user latency. In the distributed container migration approach, the edge cloud nodes can independently implement service migration strategies including the policies for suspension and migration of currently running applications. This work attempts to implement a migration framework taking these parameters into account, by which an application or cloud-hosted services are prevented from being locked into a single physical device. The service and resource management techniques are implemented using generic functions (Figure 4, right) such as: (a) service manager: to enable a required service on the node, (b) load emulation: to increase the load at given edge cloud node, (c) container manager: to keep track of container speci_c resources such as type of service and its QoS, and (d) migration controller: to start and keep track of migration. The related work and other existing literature either studies VM or implements container migration without explicitly taking the container speci_c parameters e.g. dynamic resource allocation (available processing speed, RAM and bandwidth) and container size, into account. Some studies focus on techniques of migration without analysing the effect of EC node resources, load or network variability which directly impacts the decision of effcient migration.

SYSTEM ARCHITECTURE

In mobile edge cloud, multiple mobile and stationary devices interconnected through wireless local area networks are combined to create a small cloud infrastructure at a local physical area.

An edge cloud architecture is used to decentralise (processing) power to the edges (clients/devices) of your networks. Traditionally the computing power of servers is used to perform tasks such as data minimisation or to create advanced distributed systems.



TECHNIQUES OF HETEROGENEOUS EDGE CLOUD

Distributed Computing:

Heterogeneous edge cloud relies on the distribution of computing tasks across a variety of devices and servers at the network edge. This enables parallel processing and reduces the burden on a centralized data center.

Edge Computing Infrastructure:

The deployment of edge servers and devices with varying capabilities forms the foundation of a heterogeneous edge cloud. These devices could include edge servers, IoT devices, mobile devices, and other specialized hardware.

Dynamic Resource Allocation:

Techniques for dynamically allocating resources based on the requirements of different applications and services. This involves intelligent resource management to efficiently use the available computing power, storage, and network bandwidth.

Network Optimization Device:

Implementing optimization techniques for edge networks, especially in the context of 5G. This includes strategies for load balancing, efficient routing, and minimizing latency to ensure smooth and rapid data transmission.

Mobile Edge Computing (MEC):

MEC is a key component of the heterogeneous edge cloud, focusing on bringing cloud computing capabilities to the edge of the mobile network. It enables the execution of applications on edge servers, reducing latency for mobile users.

APPLICATIONS

Internet of Things (IoT):

Heterogeneous edge cloud is well-suited for IoT applications. By distributing processing tasks to edge devices and servers, it reduces the need to send all data to a centralized cloud. This is crucial for IoT scenarios such as smart home automation, industrial IoT, and connected vehicles, where low latency and real-time processing are essential.

5G Networks:

In the context of 5G, heterogeneous edge cloud optimizes network resources. It allows for dynamic adaptation to changing demands, enhancing the performance of applications that require high bandwidth, such as augmented reality (AR), virtual reality (VR), and high-definition video streaming.

Autonomous Systems:

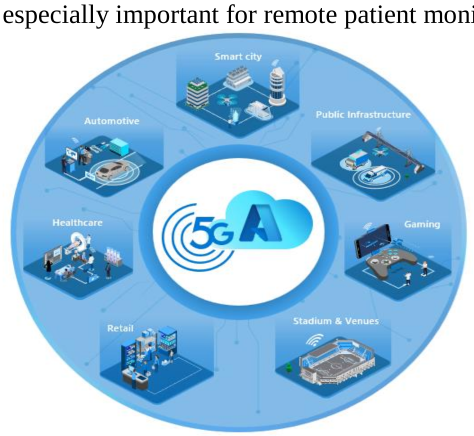
Heterogeneous edge cloud is crucial for autonomous systems like self-driving cars and drones. These systems require quick decision-making based on sensor data, and processing this data at the edge reduces communication latency, ensuring timely responses and improving overall safety.

Smart Cities:

Heterogeneous edge cloud supports various smart city applications, including traffic management, video surveillance, and environmental monitoring. Distributing computing tasks across the edge infrastructure enhances the responsiveness of these systems.

Healthcare:

In healthcare, heterogeneous edge cloud can be applied to real-time monitoring and analysis of patient data. This is especially important for remote patient monitoring, where timely responses to critical health indicators are essential.



MODULES

- Inter-node balancing.
- Data augmentation.
- Private cloud module
- Public cloud module

- Hybrid cloud module.

MODULES DESCRIPTION

INTER-NODE BALANCING :

Depending on the service dynamics, some nodes/clouds may become overloaded, provided that nodes/clouds capacity is heterogeneous. As an illustration, edge and fog data centers are small comparing to cloud data centers. it is necessary to support migration.

DATA AUGMENTATION:

We enrich the base data set with simulated data on the duration of the user tasks and the normalized workload they incur on the edge servers during that time. augmentation is a technique of artificially increasing the training set by creating modified copies of a dataset using existing data.

PRIVATE CLOUD MODULE

cloud computing environment dedicated to a single organization. Any cloud infrastructure has underlying compute resources like CPU and storage that you provision on demand through a self-service portal. In a private cloud, all resources are isolated and in the control of one organization.

Private cloud involves configuring underlying infrastructure resources yourself or on your behalf by a managed services partner such as Rackspace Technology. Public cloud configuration involves learning about the cloud provider's portal and tools. The next step is understanding how a migration is executed.

PUBLIC CLOUD MODULE

A public cloud is an IT model where public cloud service providers make computing services—including compute and storage, develop-and-deploy environments.

Cloud migration is the process of moving applications and data from one location, often a company's private, on-site ("on-premises") servers to a public cloud provider's servers, but also between different clouds.

HYBRID CLOUD MODULE

hybrid cloud is a mixed computing environment where applications are run using a combination of computing, storage.

We help you develop, execute, and evolve your hybrid cloud strategy so you achieve the right balance of resources on premises and in the cloud, simplify infrastructure management, control costs and protect data across your hybrid cloud environment.

III. CONCLUSION

This paper has proposed *ShareOn* _ a distributed shared resource framework for container migration, enabling edge cloud nodes to support system load, heterogeneity, and network Suctuations. *ShareOn* is evaluated for the real-time services deployed at ORBIT radio grid testbed, and is then simulated for San Francisco city based large-scale edge cloud network and taxicab's mobility.

The system response time and migration cost are evaluated by executing a low-latency application (ALPR) using different algorithms running at individual edge cloud nodes. In this work we determine number of migrations initiated at each edge cloud node to dynamically redistribute the resources. Also, the system performance is analyzed for various load conditions and container resource requirements. *ShareOn* is optimized with parameters such as network latency, edge cloud resources, edge cloud load, and inter-edge bandwidth for better application QoS. The system perfor- mance of ShareOn is compared with two non-migration approaches: equal-load and nearest-edge, and two migration- based approaches: bandwidth-only and processing-only.

FUTURE ENHANCEMENT

We plan to evaluate ShareOn on alternative networks and compute-aware algorithms in an edge-cloud enabled realistic outdoor testbed such as COSMOS. We plan to eval- uate ShareOn on alternative networks and compute-aware algorithms in an edge-cloud enabled realistic outdoor testbed such as COSMOS. For future work we aim to adopt an SDN infrastructure using OpenFlow controllers to evaluate migration bene_ts for 5G URLLC use cases and analyze the system response time, network and computational requirements. Section VI illustrates simulation set-up and param- eters. Results are discussed in Section VII, related work in Section VIII with a note on limitations, challenges and our future work and Section IX concludes the paper.

REFERENCES

[1] N. Abbas,Y. Zhang, A. Taherkordi, and T. Skeie, ``Mobile edge computing: A survey,'' IEEE Internet Things J., vol. 5, no. 1, pp. 450 465, Feb. 2017.

[2] S. Maheshwari, D. Raychaudhuri, I. Seskar, and F. Bronzino, ``Scalability and performance evaluation of edge cloud systems for latency constrainedapplications,'' in Proc. 3rd IEEE/ACM Symp. Edge Comput. (SEC), Oct. 2018, pp. 286 299.

[3] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, ``Collaborative mobile edge computing in 5G networks: New paradigms, scenarios, and challenges,'' IEEE Commun. Mag., vol. 55, no. 4, pp. 54 61, Apr. 2017.

[4] M. Chen and Y. Hao, ``Task of oading for mobile edge computing in software de ned ultra-dense network,'' IEEE J. Sel. Areas Commun., vol. 36, no. 3, pp. 587 597, Mar. 2018.

[5] T. Zhao, S. Zhou, X. Guo, and Z. Niu, ``Tasks scheduling and resource allocation in heterogeneous cloud for delay-bounded mobile edge computing,'' in Proc. IEEE Int. Conf. Commun. (ICC), May 2017, pp. 1 7.

[6] W. Z. Khan, E. Ahmed, S. Hakak, I. Yaqoob, and A. Ahmed, ``Edgecomputing: A survey,'' Future Gener. Comput. Syst., vol. 97, pp. 219 235, Aug. 2019.

[7] F. E. F. Samann, S. R. M. Zeebaree, and S. Askar, ``IoT provisioning QoS based on cloud and fog computing,'' J. Appl. Sci. Technol. Trends, vol. 2, no. 1, pp. 29 40, Mar. 2021.

[8] C. Cicconetti, M. Conti, and A. Passarella, ``Low-latency distributedcomputation of oading for pervasive environments,'' in Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom), Mar. 2019, pp. 1 10.

[9] J. Zhao, Q. Li, Y. Gong, and K. Zhang, ``Computation of oading and resource allocation for cloud assisted mobile edge computing in vehicular networks,'' IEEE Trans. Veh. Technol., vol. 68, no. 8, pp. 7944 7956, Aug. 2019.

[10] H.-C. Hsieh, C.-S. Lee, and J.-L. Chen, ``Mobile edge computing
platform with container-based virtualization technology for IoT appli- cations,'' Wireless Pers. Commun., vol. 102, no. 1, pp. 527 542,Sep. 2018.

[11] H. Hawilo, M. Jammal, and A. Shami, ``Exploring microservices as the architecture of choice for network function virtualization platforms,'' IEEE Netw., vol. 33, no. 3, pp. 202 210, Mar./Apr. 2019.

[12] B. Liang, M. A. Gregory, and S. Li, ``Multi-access edge computingfundamentals, services, enablers and challenges: A complete survey,'' J. Netw. Comput. Appl., vol. 199, Mar. 2021, Art. no. 103308.

[13] S. Maheshwari, S. Choudhury, I. Seskar, and D. Raychaudhuri, ``Traf c- aware dynamic container migration for real-time support in mobile edge clouds,'' in Proc. IEEE Int. Conf. Adv. Netw. Telecommun. Syst. (ANTS), Dec. 2018, pp. 1 6.

[14] P. G. Lopez, A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, and E. Riviere, ``Edge-centric computing: Visionand challenges,'' ACM SIGCOMM Comput. Commun. Rev., vol. 45, no. 5, pp. 37 42, 2015.

[15] L. Loven, E. Peltonen, E. Harjula, and S. Pirttikangas, ``Weathering thereallocation storm: Large-scale analysis of edge server workload,'' in Proc. Joint Eur. Conf. Netw. Commun. 6G Summit (EuCNC/6G Summit),Jun. 2021, pp. 317 322.

[16] C.Westphal, ``Challenges in networking to support augmented reality and
virtual reality,'' in Proc. Int. Conf. Comput., Netw. Commun. (ICNC), 2017.

[17] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, ``A survey of mobile cloud computing: Architecture, applications, and approaches,'' Wireless Commun. Mobile Comput., vol. 13, no. 18, pp. 1587 1611, Dec. 2011.

[18] A. Alomari, S. K. Subramaniam, N. Samian, R. Latip, and Z. Zukarnain,
``Resource management in SDN-based cloud and SDN-based fog computing: Taxonomy study,'' Symmetry, vol. 13, no. 5, p. 734, Apr. 2021.

[19] H. Wu, J. Chen, C. Zhou, W. Shi, N. Cheng, W. Xu, W. Zhuang, and X. S. Shen, ``Resource management in space-air-ground integrated vehicular networks: SDN control and AI algorithm design,'' IEEEWireless
Commun., vol. 27, no. 6, pp. 52 60, Dec. 2020.

[20] M. Alsaeedi, M. M. Mohamad, and A. A. Al-Roubaiey, ``Toward adaptive and scalable OpenFlow-SDN ow control: A survey,'' IEEE Access, vol. 7, pp. 107346 107379, 2019.