Runtime Model for Distributed Workload in the Edge-to-Cloud Continuum

Ivan Čilić, Ivana Podnar Žarko
University of Zagreb, Faculty of Electrical Engineering and Computing
Department of Telecommunications
Zagreb, Croatia
ivan.cilic@fer.hr, ivana.podnar@fer.hr

***Abstract*—Edge-to-Cloud Continuum (ECC) optimizes IoT data processing by executing it closer to the data sources. The hierarchical structure of ECC spans from resource-constrained IoT devices to high-performance cloud servers, providing a distributed IoT environment with lower network congestion, improved response times and enhanced security compared to cloud-based IoT solutions. Despite its advantages, ECC faces challenges due to its heterogeneous and dynamic nature that has to deal with uneven distribution of data sources. To address these challenges, in this doctoral research, we propose a runtime model for distributed workload in ECC. The model describes the behavior of data sources in ECC, the services required to process the generated data, and the underlying topology of ECC that hosts these services. This model serves as the basis for the implementation of the ECC workload simulator which provides insight into the data dynamics within the edge computing nodes and the ability of the processing services to handle the incoming workload. By simulating the workload and processing in a distributed ECC environment, the simulator can help optimize topology design, service scheduling, and data routing decisions.**

***Index Terms*—Internet of Things, Edge Computing, Data Density, Workload**

I. Introduction and related work

The Edge-to-Cloud Continuum (ECC) concept aims to unburden the cloud by facilitating the processing of IoT data as close to the data sources as possible. The processing can occur on devices situated on the edge of the network, either nearby or further away, thereby significantly reducing the amount of data that needs to be transmitted to the cloud. The ECC organizes devices in a hierarchical structure, beginning with IoT devices at the bottom layer. These IoT devices host sensors and actuators but have limited resources. They connect to nearby gateway nodes and local devices in the far-edge layer. Far-edge nodes are located close to IoT devices, often within the same local network or just one hop away. Above the far-edge layer is the near-edge layer, comprised of more powerful compute nodes, such as a local micro-cloud with several server racks. At the topmost layer is the cloud, situated in data centers and possessing virtually limitless resources. By expanding the cloud with additional compute nodes in the ECC, the processing and storage capabilities become more accessible and more efficient to end devices. The ECC approach presents several benefits for IoT solutions, including reduced Internet traffic, shorter response times, enhanced security with privacy controls, and decreased operational costs.

This work has been supported by Croatian Science Foundation under the project IP-2019-04-1986.

The ECC environment is marked by constant changes in the status and positions of devices and nodes providing computation and storage resources to IoT devices. More importantly, the distribution of devices and the data they produce across this continuum is not uniform. In the context of modern IoT services, where substantial data volumes are generated and need near real-time processing, novel challenges arise. Given the incoming workload and the services required to handle it, the following questions need to be answered:

1) How to design the ECC's architecture and topology?
2) How to effectively schedule services within the ECC?
3) How to route the data to the appropriate services?

All three of these challenges are influenced by the dynamic and dispersed nature of the workload within the ECC. To address this complexity, within this doctoral research, we propose a runtime model for distributed workload in the ECC. The model holds the information on where the data sources (IoT devices) are placed in the ECC, the amount of data they generate considering the size and frequency of data generation, and the data being received and processed by services in the ECC hierarchy.

The model will serve as an input to the *ECC workload simulator*. Based on the given model, the simulator would provide information on the amount of data that is generated, collected, and processed within a specific edge computing environment or location. Also, it will show how the system design handles the incoming workload and the time between the data being generated and the final step in the processing pipeline. This information can help to design the topology of the ECC by placing more nodes with higher resources in locations with high workloads, and to decide where to place additional service instances to support the processing of larger amounts of data. Finally, it can help improve data routing to the processing services to equally balance node and service loads, and help in designing and optimizing edge computing architectures to achieve a desired balance between local processing and cloud-based analysis.

Our work on modeling distributed workload in ECC is closely correlated to the workload prediction problem at the edge. Works in this field often try to describe the workload and to predict how the workload will change over time due to

the moving of the data sources in the network or an increase in the generation frequency. To compare, our idea is to model the current data workload considering various data source characteristics and the distributed nature of ECC, and prediction is one of the possible usages of this model. The authors in [3] propose a location-aware workload prediction in MEC based on LSTM. They input historical workload information into LSTM to predict the workload for each Edge Data Center (EDC). Finally, they evaluate the proposed algorithms on a use case with real mobility traces in a cellular network. In comparison, we take into account the heterogeneity of the data sources, not only the number of the sources, but also the data source characteristics such as data size and generation frequency. Another similar approach can be found in [1]. The authors propose an approach to model input load for Smart City services. They analyze different Smart City services based on data generation frequency, number of data sources and single source distribution, to obtain the overall input load distribution for a given service. The main goal of their proposed approach is to provide insight into the input rates of the service which can be expected in real-world service deployments, so that IoT solutions can be evaluated with regard to their performance in real-world environments. Similarly, our goal is also to model the input workload for a given service, with the main difference being that we are taking into account the distributed nature of the ECC environment, so we are not only modeling the overall load, but the load in a particular location and the ability of a processing service to handle it.

II. Methodology and Conceptual Approach

Our research is divided into the following stages:

A. Analysis of workload in ECC

In our previous work, we have conducted research in the field of data routing in ECC [2]. The main idea was to implement a data proxy on each node in ECC to ensure continuous data delivery from data sources to data processing services while maintaining high QoS for the data sources. During the research, we realized that both routing and scheduling decisions largely depend on the behavior and position of the data sources. Therefore, we have conducted an analysis of the ECC workload to obtain the parameters needed to model it.

B. Runtime model for distributed workload in ECC

The proposed model is an abstraction of a running system that contains the following information: (i) the topology of the ECC, i.e., the distribution of nodes in the network and the set of resources they hold, (ii) the data sources, their data generation functions, and the gateway node they are connected to, (iii) the distribution of processing services and the performance model of the services, and (iv) the connection between the data source and the target processing service.

C. Implementation of an ECC workload simulator

To streamline the testing and evaluation of ECC implementations while accommodating various incoming workloads, we plan to develop an ECC workload simulator. This simulator will provide the flexibility to define custom ECC topologies and simulate the behavior of different data sources. It also enables the integration of data processing services and replicates the entire ECC workflow — from initial data generation to subsequent data processing nodes. The simulator allows users to simulate changes in workloads, a feature particularly useful for evaluating dynamic scenarios that can be specified via custom-defined functions.

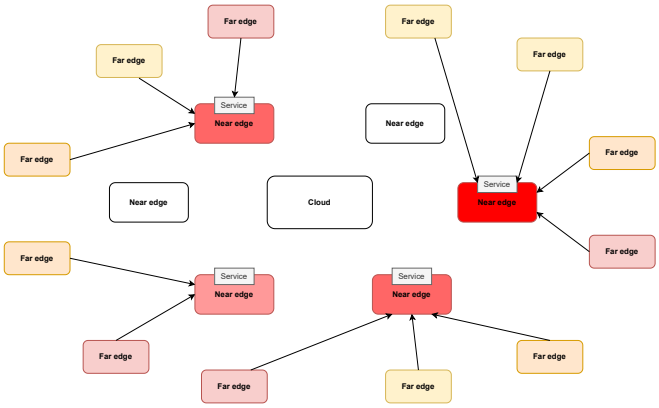

Fig. 1. Abstraction of an ECC workload simulator.

Figure 1 shows an example of how the simulator works. It aggregates the incoming workloads from data sources on a particular gateway (far edge) node and simulates the movement of the workload through the ECC topology to the processing service. Finally, the simulator provides information about the load for each processing service, the amount of data successfully processed, and the time it takes from data generation to successful processing.

D. Use case for verification of the simulator

The final phase of the research is to identify and implement a real-world IoT use case that demonstrates the applicability of the simulator to optimize service scheduling in ECC. As our recent work relates to federated learning, our idea is to simulate an FL inference pipeline to optimize the placement of FL inference services by reducing the time from data generation to prediction made by the inference service.

References

[1] Antonić, A., Marjanović, M., Žarko, I.P.: Modeling aggregate input load of interoperable smart city services. In: Proceedings of the 11th ACM International Conference on Distributed and Event-Based Systems. p. 34–43. DEBS '17, Association for Computing Machinery, New York, NY, USA (2017). https://doi.org/10.1145/3093742.3093928, https://doi.org/10.1145/3093742.3093928

[2] Čilić, I., Žarko, I.P.: Adaptive data-driven routing for edge-to-cloud continuum: A content-based publish/subscribe approach. In: González-Vidal, A., Mohamed Abdelgawad, A., Sabir, E., Ziegler, S., Ladid, L. (eds.) Internet of Things. pp. 29–42. Springer International Publishing, Cham (2022)

[3] Nguyen, C., Klein, C., Elmroth, E.: Multivariate lstm-based location-aware workload prediction for edge data centers. In: 2019 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID). pp. 341–350 (2019). https://doi.org/10.1109/CCGRID.2019.00048