EdgeScaler: Smart (Auto-)Scaling for the 5G Edge

Lauren Trinks, Bilal Saleem, and Muhammad Shahbaz
Purdue University

Introduction and Motivation. The 5G technology is rapidly expanding and pushing across user experience and design boundaries. One significant advancement is deploying a 5G core as microservices on edge clouds, strategically positioned close to end users [4]. These 5G edge clouds facilitate applications with high computational demands to offload processing tasks, resulting in millisecond scale latencies per UE request [1]. Notably, the minimal latency at the edge cloud highlights a stark contrast with centralized data centers, where network transit time typically dominates the client request-response cycle—making the overhead associated with scheduling, scaling, and load balancing in edge cloud environments more pronounced than in centralized data centers [2].

While Kubernetes remains an industry standard across both centralized data centers and edge clouds, its design and optimization primarily cater to the workloads running on the cloud data center architectures [3]. Our study delves into Kubernetes' Horizontal Pod Autoscaling (HPA) performance concerning meeting 5G core Service Level Optimizations (SLOs). Additionally, we present a proof of concept (PoC) for a novel autoscaling solution, EdgeScaler, drawing inspiration from emerging trends in service-mesh architectures (Figure 1). EdgeScaler is tailored to integrate seamlessly with any 5G core operating on Kubernetes; it aims to enhance adherence to response latency SLOs through enhanced flexibility and transparency in autoscaling behaviors, including the use of machine learning (ML) within the decision-making process.

• *Goals:* We need a scaling scheme for the 5G edge that can bring up/down resources (e.g., pods) efficiently without under-utilizing (i.e., resource wastage) or over-utilizing them (i.e., leading to high tails).

• *Challenges:* The lack of support and flexibility in existing scaling schemes (e.g., k8s HPA). These are tailored for the cloud environments and rely on fixed thresholds (i.e., %CPU) for scaling up instances, thus resulting in long tails or idle resources.

Design Overview. EdgeScaler consists of four key components: (1) Metrics Collector, capable of gathering metrics from various sources (e.g., pods and kernel); (2) Metrics Modification, for creating new heuristics (e.g., for smoothing and forecasting); (3) Decision Maker, to make decisions based on these heuristics; and 4) Executor, responsible for interacting with k8s to scale instances up or down.

• *EdgeScaler's LSTM-based Predictor.* It employs a machine-learning model, specifically Long Short Term Memory (LSTM), to make predictions. The model consumes a timeseries of

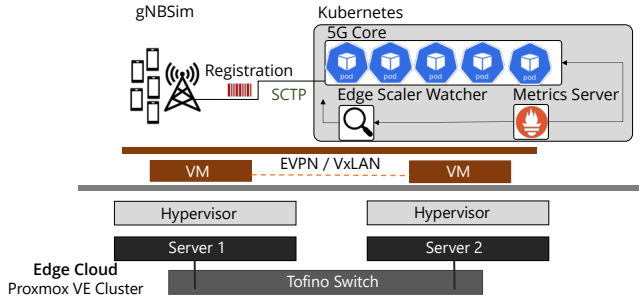

Figure 1: Design of an EdgeScaler-based k8s Cluster.

CPU usage information over a given window and forecasts utilization 15 minutes into the future. For our specific model, we used a learning rate of 0.0001, 100 epochs, and a root mean squared error (RMSE) of 0.0184. Predictions generated by the model are used as the current metric value for the scaling algorithm used in k8s HPA.

Preliminary Results. Figure 2 illustrates the advantages of EdgeScaler. By enabling forecasting, scaling can be executed before the load increases. When traffic surges, more CPUs are distributed across sufficient pods to mitigate the impact of load spikes, reducing average CPU utilization per pod. With precise and timely predictions, early scaling is expected to maintain the CPU utilization target for edge workloads.

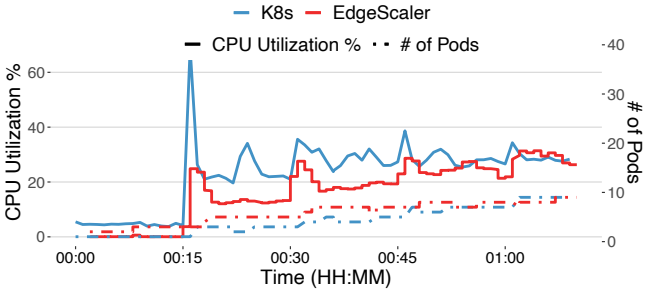

Figure 2: k8s HPA vs. EdgeScaler LSTM HPA.

REFERENCES

[1] Mukhtiar Ahmad, Syed Usman Jafri, Azam Ikram, Wasiq Noor Ahmad Qasmi, Muhammad Ali Nawazish, Zartash Afzal Uzmi, and Zafar Ayyub Qazi. 2020. A Low Latency and Consistent Cellular Control Plane. In *ACM SIGCOMM*.

[2] Batyr Charyyev, Engin Arslan, and Mehmet Hadi Gunes. 2020. Latency Comparison of Cloud Datacenters and Edge Servers. In *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*. 1–6. https://doi.org/10.1109/GLOBECOM42002.2020.9322406

[3] Google Cloud. [n. d.]. What is Kubernetes? ([n. d.]). https://cloud.google.com/learn/what-is-kubernetes last accesed 4/2024.

[4] Oguz Sunay Larry Peterson and Bruce Davie. 2022. *Private 5G: A Systems Approach.* Systems Approach LLC.