# Poster: KUT: Towards Lightweight On-path Network Assessment for Edge Orchestration

Geng, Wei; Altas, Oguz Kagan; Guzman, David; Bartolomeo, Giovanni; Mohan, Nitinder; Ott, Joerg

**DOI**
10.1145/3765515.3771746

**Publication date**
2025

**Document Version**
Final published version

**Published in**
CoNEXT 2025 - Proceedings of the 21st International Conference on Emerging Networking EXperiments and Technologies

POSTER

# Poster: KUT: Towards Lightweight On-path Network Assessment for Edge Orchestration

**WEI GENG**, Technical University of Munich, Munich, Bayern, Germany

**OGUZ KAGAN ALTAS**, Technical University of Munich, Munich, Bayern, Germany

**DAVID E GUZMÁN**, Technical University of Munich, Munich, Bayern, Germany

**GIOVANNI BARTOLOMEO**, Technical University of Munich, Munich, Bayern, Germany

**NITINDER MOHAN**, Delft University of Technology, Delft, Zuid-Holland, Netherlands

**JÖRG OTT**, Technical University of Munich, Munich, Bayern, Germany



**Published:** 01 December 2025

**Citation in BibTeX format**

CoNEXT '25:The 21st International Conference on emerging Networking EXperiments and Technologies
*December 1 - 4, 2025*
*Hong Kong, Hong Kong*

**Conference Sponsors:**
SIGCOMM

CoNEXT '25: Proceedings of the 21st International Conference on emerging Networking EXperiments and Technologies (December 2025)
https://doi.org/10.1145/3765515.3771746
ISBN: 9798400721915

# Poster: KUT: Towards Lightweight On-path Network Assessment for Edge Orchestration

Wei Geng
wei.geng@tum.de
Technical University of Munich
Munich, Germany

Oguz Kagan Altas
oguz.altas@tum.de
Technical University of Munich
Munich, Germany

David Guzman
david.guzman@tum.de
Technical University of Munich
Munich, Germany

Giovanni Bartolomeo
giovanni.bartolomeo@tum.de
Technical University of Munich
Munich, Germany

Nitinder Mohan
n.mohan@tudelft.nl
Delft University of Technology
Delft, Netherlands

Joerg Ott
ott@in.tum.de
Technical University of Munich
Munich, Germany

## Abstract

Edge computing orchestration faces significant challenges due to resource constraints, highly distributed topologies, and dynamic network conditions. The discrepancy between theoretical and actual runtime performance often leads to suboptimal deployment decisions. This discrepancy is severe in clustered deployments, as existing tools either saturate network links during testing or lack proactive assessment capabilities. None of these approaches accurately predicts service co-locating compatibility in real-world scenarios. We present KUT (Konnectivity Under Test), a lightweight network assessment framework designed specifically for edge environments, combining service-specific traffic simulation with periodical monitoring to provide accurate compatibility assessments without starving co-located services. KUT enables periodic background assessments that inform orchestration decisions while consuming minimal resources.

## CCS Concepts

• **Networks** → **Network measurement**; **Network monitoring**; *Cloud computing*; *Network management*.

## Keywords

Edge Computing, Network Assessment, Orchestration, Compatibility Testing, Lightweight Monitoring, Traffic Simulation

**ACM Reference Format:**
Wei Geng, Oguz Kagan Altas, David Guzman, Giovanni Bartolomeo, Nitinder Mohan, and Joerg Ott. 2025. Poster: KUT: Towards Lightweight On-path Network Assessment for Edge Orchestration. In *Proceedings of the 21st International Conference on emerging Networking EXperiments and Technologies (CoNEXT '25), December 1–4, 2025, Hong Kong, Hong Kong*. ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3765515.3771746


*CoNEXT '25, Hong Kong, Hong Kong*

ACM ISBN 979-8-4007-2191-5/2025/12
https://doi.org/10.1145/3765515.3771746

## 1 Introduction

Mobile and weak devices such as monitoring cameras, smart home sensors, and even mobile drones and phones offload their computing tasks to nearby edge servers such as energy-efficient and dedicated hardware accelerators (e.g., Nvidia Jetson Series). This paradigm, known as edge computing, improves service responsiveness by bringing computation closer to end devices [13], compared to traditional cloud computing. However, network interference, device mobility, and heterogeneous hardware lead to a critical gap between expected and actual application performance, particularly for mission-critical, latency-sensitive applications [11], which commonly prevents effective service orchestration [12].

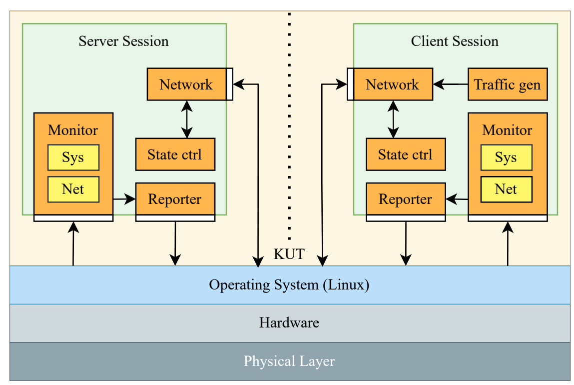


**Figure 1: KUT Architecture Overview**

The current host-to-host, pipe-based network, primarily built on TCP/UDP protocols, struggles to achieve better performance visibility. We cannot fully transition to new architectures, such as Information-Centric Networking, because most systems and applications still rely on the traditional TCP/UDP-based stack [7]. Existing network assessment tools are ill-suited for this task. Active probes like iPerf [6] saturate the bandwidth, starving co-located services. On the other hand, passive monitors like Wireshark [14] cannot predict network performance with certain traffic patterns. A tool that performs periodic, lightweight, and proactive assessments without starving production workloads is needed.

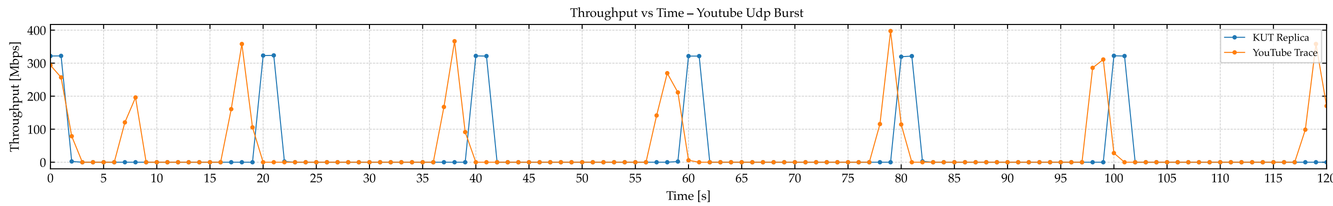


**Figure 2: A YouTube UDP Burst Traffic Simulation Showcase Using KUT**

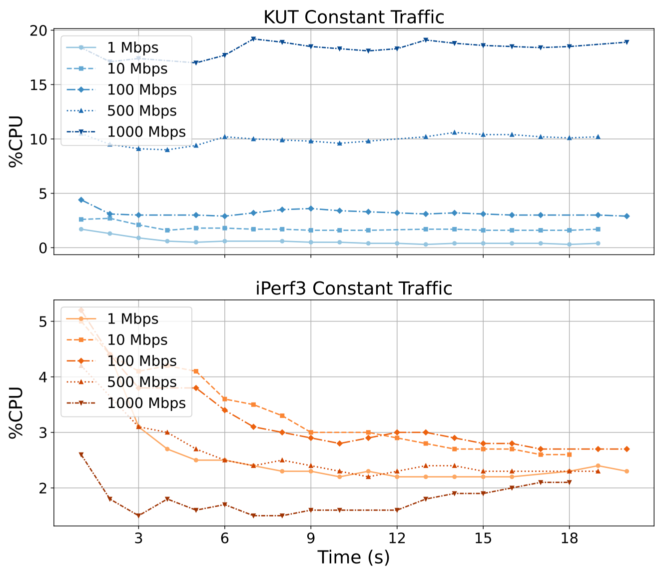


**Figure 3: CPU Usage between KUT and iPerf3**

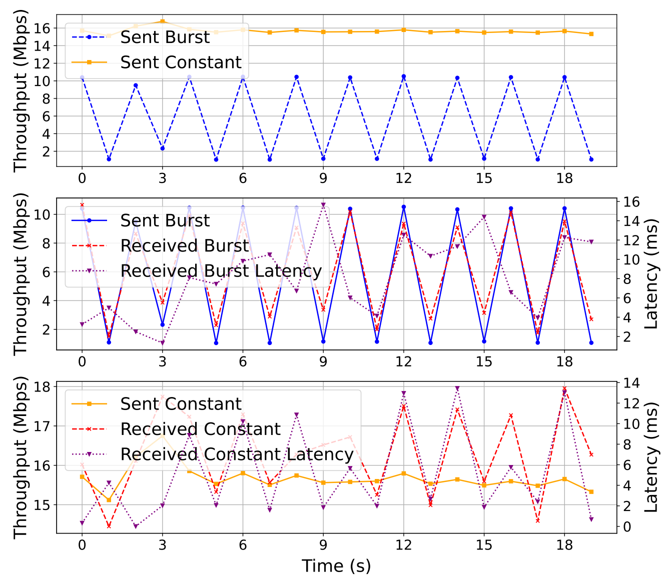


**Figure 4: Sent/Received Traffic (Burst/Constant)**

To bridge this gap, we introduce KUT, a lightweight network assessment framework designed for edge orchestration. At its core, KUT models application behavior by abstracting it into fundamental traffic parameters: packet size and transmission timing. This abstraction allows KUT to simulate the key traffic characteristics of various applications (e.g., video streaming, VoIP) by generating patterns like burst, ramp-up/down, and sinusoidal. Given an edge environment (e.g., Kubernetes [1]), there is a controller that decides when and which node pair to be evaluated by launching a pair of client and server (CS) instances (by containers) on the selected nodes. The CS pair then runs the traffic according to the specified configuration. Finnaly the controller collects the results from the server and dumps them into csv files for further analysis. These results can be used to inform orchestration decisions, such as service placement and resource allocation. Figure 2 showcased a YouTube-like UDP burst traffic pattern generated by KUT. This adaptability ensures KUT can assess a wide range of services.

This qualitative modeling approach is essential to KUT's *non-intrusive and lightweight character*. By generating representative traffic patterns instead of replaying full traffic traces, KUT delivers meaningful results by capturing key traffic characteristics with *minimal computational cost and network overhead*. It eliminates the need to capture, store, and replay every single packet, which consumes significant resources, allowing it to operate as an unobtrusive background service. To further ensure a minimal footprint and ease of deployment, as shown in Figure 1, KUT features a modular, two-layer architecture with a control layer for test management and a data layer for traffic generation, transmission, and metadata storage. Its core components are containerized as Docker images and built as CLI tools, simplifying deployment in heterogeneous edge environments, suitable for integration with a wide range of existing orchestration frameworks.

In summary, KUT contributes a novel, lightweight network assessment methodology that combines service-specific traffic simulation with periodic monitoring to inform edge orchestration. This benefits two usecases: (1) proactive assessment of service compatibility in dynamic edge environments, this means orchestration systems can make informed decisions based on realistic network performance predictions rather than theoretical models; and (2) lightweight background connectivity monitoring over time without starving co-located services, enabling continuous performance visibility as a background service with minimal resource consumption. Our preliminary results validate KUT's ability to simulate typical traffic patterns and reveal network dynamics critical for improving service compatibility. KUT is available as an open-source project at https://github.com/cm-edge/konnectivity-under-test.

## 2 Preliminary Results

We compared KUT with iPerf3 on two Raspberry Pi 4B nodes. Figure 3 shows that KUT has comparable CPU overhead below 100 Mbps. While its usage is higher beyond this rate due to the computational cost of simulating traffic patterns, which is missing for iPerf, it remains under 20% at 1000 Mbps, confirming its lightweight design.

Besides above mentioned YouTube-like UDP burst traffic (Figure 2), we co-located burst and constant traffic flows (Figure 4) and observed mutual interference: burst traffic's bandwidth was flattened, while constant traffic's was warped. Both experienced similar, fluctuating latency. These preliminary results highlight KUT's potential for uncovering complex network dynamics, such as the impact of bandwidth-heavy tasks on latency-sensitive services in edge environments.

## 3 Future Work

***Hybrid Active-Passive Monitoring.*** We will evolve KUT into a hybrid monitoring system, integrating eBPF-based passive analysis with our active testing capabilities. This synergy will correlate simulated traffic with real-world network dynamics to build a comprehensive network profile, capturing both expected behaviors and emergent anomalies. The resulting insights will enable more accurate, network-aware orchestration in dynamic edge environments. KUT can be evolved into a hybrid monitoring system, integrating eBPF-based passive analysis with active testing. This system will correlate simulated traffic with real-world network dynamics to profile expected behaviors, enabling more accurate, network-aware orchestration in dynamic edge environments.

***Lightweight Integration with Edge Orchestrators.*** KUT will be packed as a modular component for seamless integration with edge orchestration frameworks such as KubeEdge[2], K3S[3] and Oakestra [5]. To ensure efficient operation in resource-constrained edge environments, we will develop a lightweight metrics framework that avoids the high overhead of standards like OpenMetrics[4]. This integrated solution will offer operators an easy-to-deploy tool for incorporating network affinity assessments into their workflows with a minimal resource footprint.

***Resource Concentrated Environments.*** Decentralized applications maintain consistent computing states for decision-making. Consensus algorithms maintain this state [10]. Peer concentration is standard across deployments of decentralized applications, i.e., nodes deployed in common sites [9]. This resource concentration poses stricter guarantees for resource prediction and service compatibility evaluation. Therefore, we will test KUT's prediction performance in highly clustered networks, i.e., datacenter networks. This evaluation may be based on the insights for resource concentration across decentralized applications found in [8, 9].

## References


[1] Kubernetes Authors. Accessed: September 2025. Kubernetes. https://kubernetes.io.

[2] KubeEdge Project Authors. Accessed: September 2025. A Kubernetes Native Edge Computing Framework. https://kubeedge.io.

[3] K3S Project Authors. Accessed: September 2025. Lightweight Kubernetes. https://k3s.io.

[4] OpenMetrics Authors. Accessed: September 2025. OpenMetrics. https://openmetrics.io.

[5] Giovanni Bartolomeo, Mehdi Yosofie, Simon Bäurle, Oliver Haluszczynski, Nitinder Mohan, and Jörg Ott. 2023. Oakestra: A lightweight hierarchical orchestration framework for edge computing. In *2023 USENIX Annual Technical Conference (USENIX ATC 23)*. 215–231.

[6] ESnet. [n. d.]. iPerf - The ultimate speed test tool for TCP, UDP and SCTP. https://iperf.fr/. Accessed: September 2025.

[7] Wei Geng, Yulong Zhang, Dirk Kutscher, Abhishek Kumar, Sasu Tarkoma, and Pan Hui. 2023. SoK: distributed computing in ICN. In *Proceedings of the 10th ACM Conference on Information-Centric Networking*. 88–100.

[8] David Guzman, Dirk Trossen, Mike McBride, and Xinxin Fan. 2022. Insights on Impact of Distributed Ledgers on Provider Networks. (2022), 3–17.

[9] David Guzman, Dirk Trossen, and Joerg Ott. 2024. Communication Cost for Permissionless Distributed Consensus at Internet Scale. (2024), 29–34. doi:10.1145/3607504.3609288

[10] Heidi Howard. 2014. Distributed consensus revised. *University of Cambridge* UCAM-CL-TR-935 (2014).

[11] Andrew Jeffery, Heidi Howard, and Richard Mortier. 2021. Rearchitecting kubernetes for the edge. In *Proceedings of the 4th International Workshop on Edge Systems, Analytics and Networking*. 7–12.

[12] Bruce Spang, Shravya Kunamalla, Renata Teixeira, Te-Yuan Huang, Grenville Armitage, Ramesh Johari, and Nick McKeown. 2023. Sammy: Smoothing video traffic to be a friendly internet neighbor. In *Proceedings of the ACM SIGCOMM 2023 Conference*. 754–768.

[13] Xiaofei Wang, Yiwen Han, Victor CM Leung, Dusit Niyato, Xueqiang Yan, and Xu Chen. 2020. Convergence of edge computing and deep learning: A comprehensive survey. *IEEE communications surveys & tutorials* 22, 2 (2020), 869–904.

[14] Wireshark Foundation. [n. d.]. Wireshark: The world's leading network protocol analyzer. https://www.wireshark.org. Accessed: September 2025.