Chair of Connected Mobility
TUM School of Computation, Information and Technology
Technical University of Munich

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

Wei Geng[1], Oguz Kagan Altas[1], David Guzman[1], Giovanni Bartolomeo[1], Nitinder Mohan[2], and Jörg Ott[1]

1. The Challenge

Edge computing brings computation closer to devices, but faces critical gaps:

▶ Current Tools Fall Short:

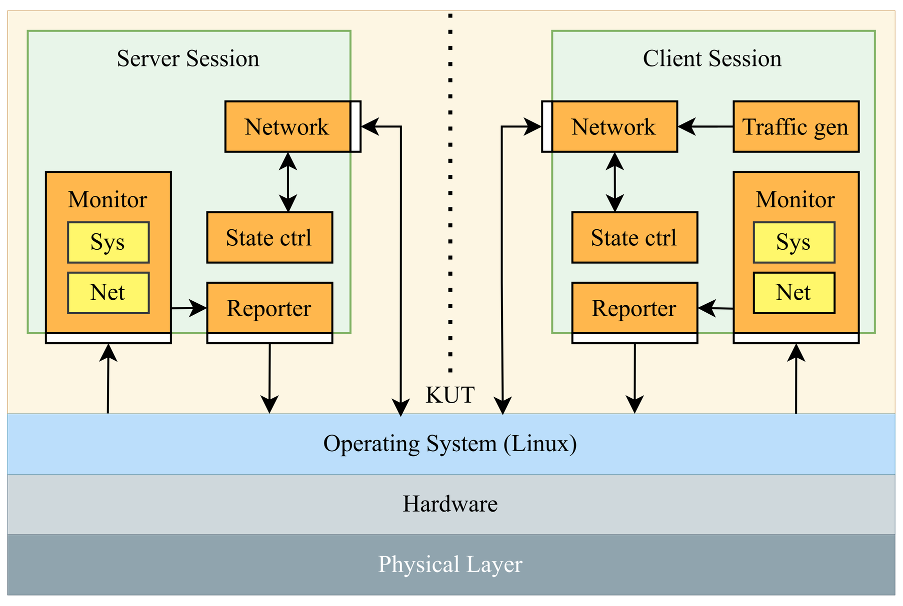
- **Active probes** (e.g., iPerf) saturate bandwidth, starving co-located services
- **Passive monitors** (e.g., Wireshark) cannot predict performance with specific traffic patterns

Neither approach accurately assesses service compatibility

▶ Solution: KUT

Periodic, lightweight, and proactive assessments without interfering with production workloads.

2. KUT Architecture



▶ Modular Two-Layer Design:

- **Control Layer:** Orchestrates test scheduling and result collection
- **Data Layer:** Generates traffic, transmits data, stores metadata
- **Containerized:** Easy deployment in heterogeneous edge environments

cm-edge/konnectivity-under-test

GitHub Repository

Chair of Connected Mobility

Chair Homepage

3. Key Features & Use Cases

▶ Lightweight Implementation:

- Qualitative traffic modeling vs. full trace replay
- Minimal computational cost & network overhead
- CPU usage <20% even at 1000 Mbps

▶ Configurable Traffic:

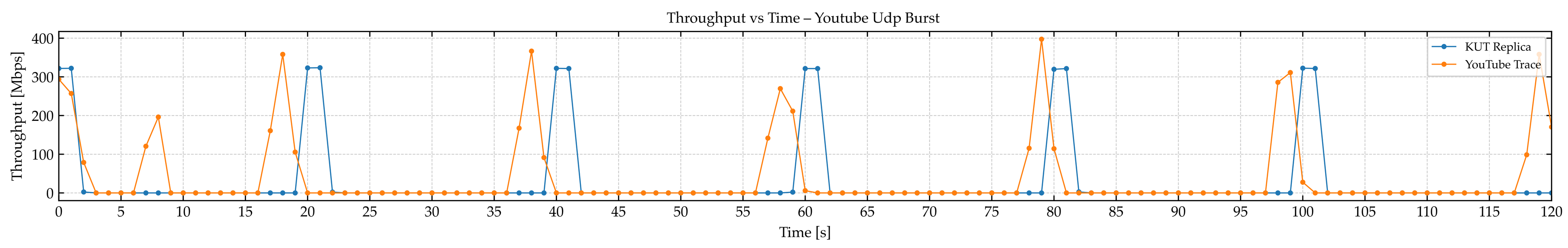
- Burst, ramp-up/down, sinusoidal, constant
- Simulates video streaming, VoIP, file transfers

▶ Primary Use Cases:

- **Proactive Compatibility:** Assess service compatibility before deployment based on realistic network predictions
- **Continuous Monitoring:** Lightweight background connectivity monitoring without starving co-located services

4. Traffic Pattern Simulation

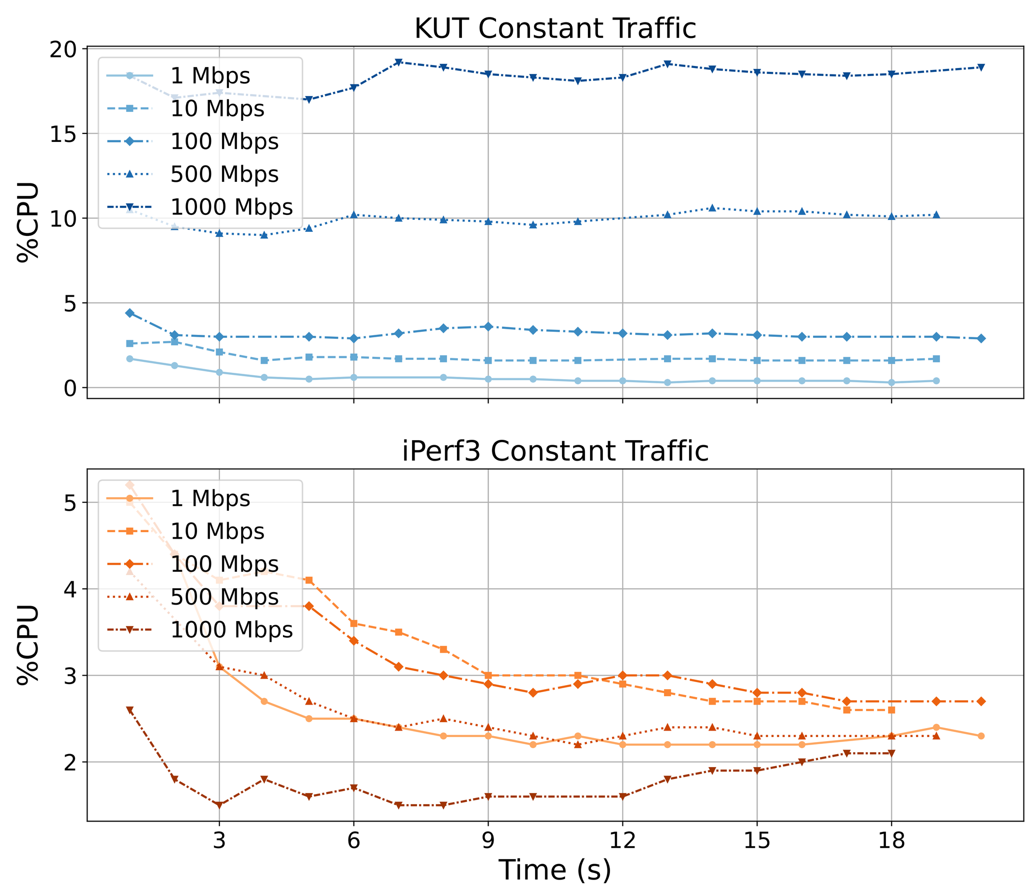
▶ YouTube-like UDP Burst Traffic Pattern:



KUT captures key traffic characteristics by modeling fundamental parameters (**packet size & transmission timing**) rather than replaying full traces. This approach provides **meaningful insights** with **minimal resource consumption**.

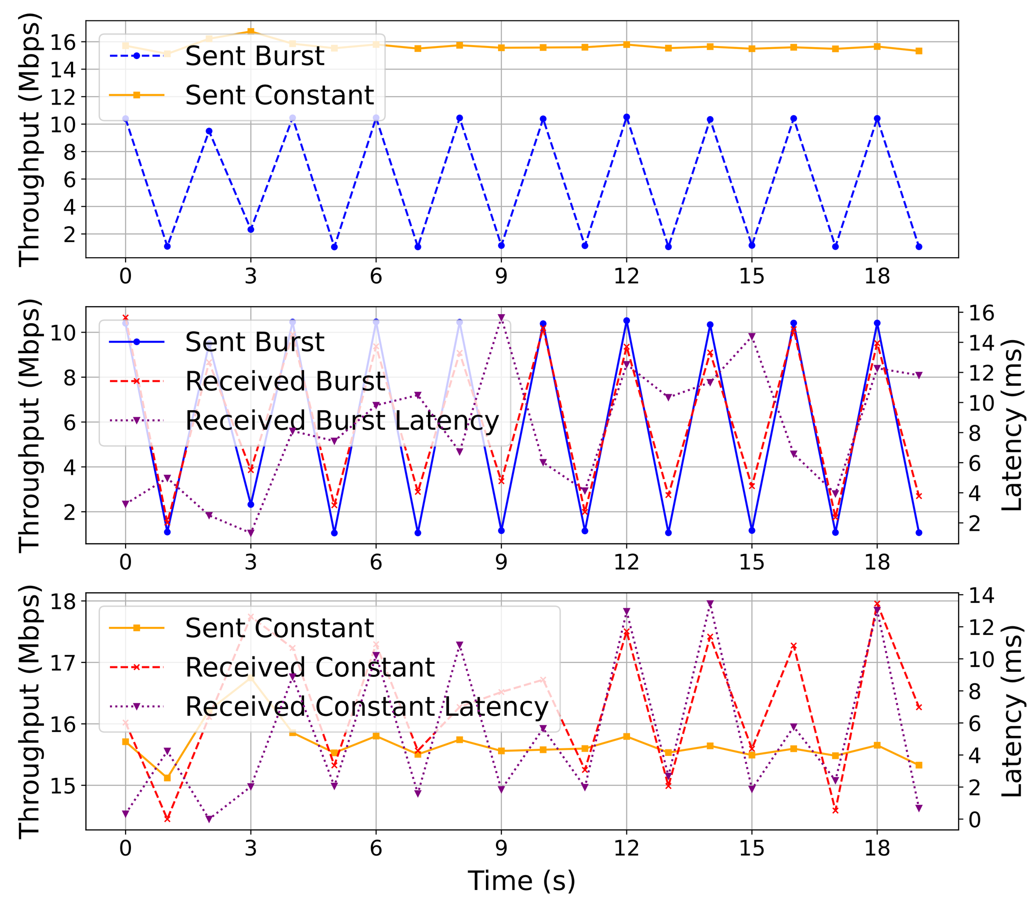
5. Preliminary Results

▶ CPU Usage: KUT vs. iPerf3



Tested on Raspberry Pi 4B nodes. KUT maintains **<10% CPU overhead below 100 Mbps** and **<20% at 1000 Mbps**, confirming lightweight design suitable for resource-constrained edge devices.

▶ Service Co-location Insights



When co-locating burst and constant flows, KUT reveals **mutual interference**: burst bandwidth flattens, constant traffic warps, and both experience fluctuating latency—critical insights for service placement decisions.

6. Future Work

▶ Next Steps:

- **Hybrid Monitoring:** Integrate eBPF-based passive analysis with active testing
- **Orchestrator Integration:** Seamless integration with KubeEdge, K3S, Oakestra platforms
- **Datacenter Evaluation:** Test prediction in resource-concentrated environments such as hyperscalers

[1]Technical University of Munich, Germany
[2]Delft University of Technology, Netherlands

Contact: wei.geng@tum.de

CoNEXT '25 | Hong Kong SAR | December 1–4, 2025

Key References:

- **Edge Computing:** Wang et al. "Convergence of edge computing and deep learning" (Proc. IEEE 2020)
- **Network Tools:** iPerf - Network performance measurement tool
- **Orchestration:** Spang et al. "Sammy: Service orchestration" (NSDI 2023)
- **Frameworks:** Kubernetes, KubeEdge, K3S, Oakestra