Latency-Aware Scheduling in Edge Networks

Kevin James
Dept. of Engineering
Some State University
Some City, A State, USA
KevJames@ssu.edu

Jane Smith
Dept. of Engineering
Some State University
Some City, A State, USA
JaneSmith@ssu.edu

***Abstract*—Edge computing environments are highly dynamic, with resource constraints and network variability posing significant challenges to efficient task scheduling. This paper presents a latency-aware scheduling algorithm designed to improve performance for latency-sensitive applications deployed at the network edge. By incorporating real-time network profiling and predictive modeling, our approach dynamically prioritizes tasks based on latency sensitivity and resource availability. Simulation results demonstrate reductions in task queuing time, improved throughput under burst loads, and superior fairness compared to baseline schedulers. This work contributes a lightweight yet adaptive scheduling framework suitable for constrained edge deployments in IoT and AI scenarios.**

I. Introduction

As computation moves closer to the data source, edge computing has emerged as a key enabler for real-time and context-aware applications. Domains such as autonomous vehicles, smart healthcare, and industrial monitoring demand stringent latency requirements that traditional cloud-centric infrastructures cannot meet. To address these demands, edge devices must efficiently manage computation, often under variable network conditions and limited resources.

A core challenge in this setting is scheduling: deciding which tasks to execute, defer, or offload in a way that minimizes response time and maximizes utilization. Conventional schedulers, which often rely on fixed priority queues or round-robin schemes, are ill-suited to the unpredictability of edge workloads.

In this paper, we propose a latency-aware scheduling framework that adapts in real time to changing workload characteristics and system conditions. Our scheduler continuously evaluates the delay tolerance of tasks and makes informed decisions about task execution order and placement. We embed a lightweight profiling mechanism that tracks queuing delays, CPU contention, and network jitter, allowing the system to predict scheduling bottlenecks and respond preemptively.

The remainder of the paper is organized as follows: Section II surveys prior work in edge scheduling and latency modeling. Section III details the architecture and algorithmic components of our framework. Section IV provides performance evaluations in simulated edge topologies. Section V concludes with discussion and future directions.

II. Related Work

Edge task scheduling has been addressed in various contexts, including real-time systems, fog computing, and distributed AI platforms. Existing solutions typically fall into two categories: static priority-based schedulers and dynamic load balancers.

Static strategies, such as earliest-deadline-first (EDF) and fixed-priority preemptive scheduling, offer predictability but lack flexibility under fluctuating conditions. Dynamic approaches, including heuristic-based and reinforcement learning schedulers, aim to adapt to workload changes but often incur high computational overhead or require extensive training time.

Recent works like EDGEWISE [1] and FogSched [2] have introduced latency-conscious heuristics, but these are often tailored for specific domains (e.g., multimedia streaming) or lack general applicability. Moreover, many prior studies assume homogeneous resources and overlook network variability, a critical factor in real-world edge environments.

Our approach differs in that it combines latency prediction, resource profiling, and adaptive task reordering into a single, lightweight module designed for heterogeneous, bandwidth-variable environments. We explicitly target scenarios with tight response constraints and minimal central coordination.