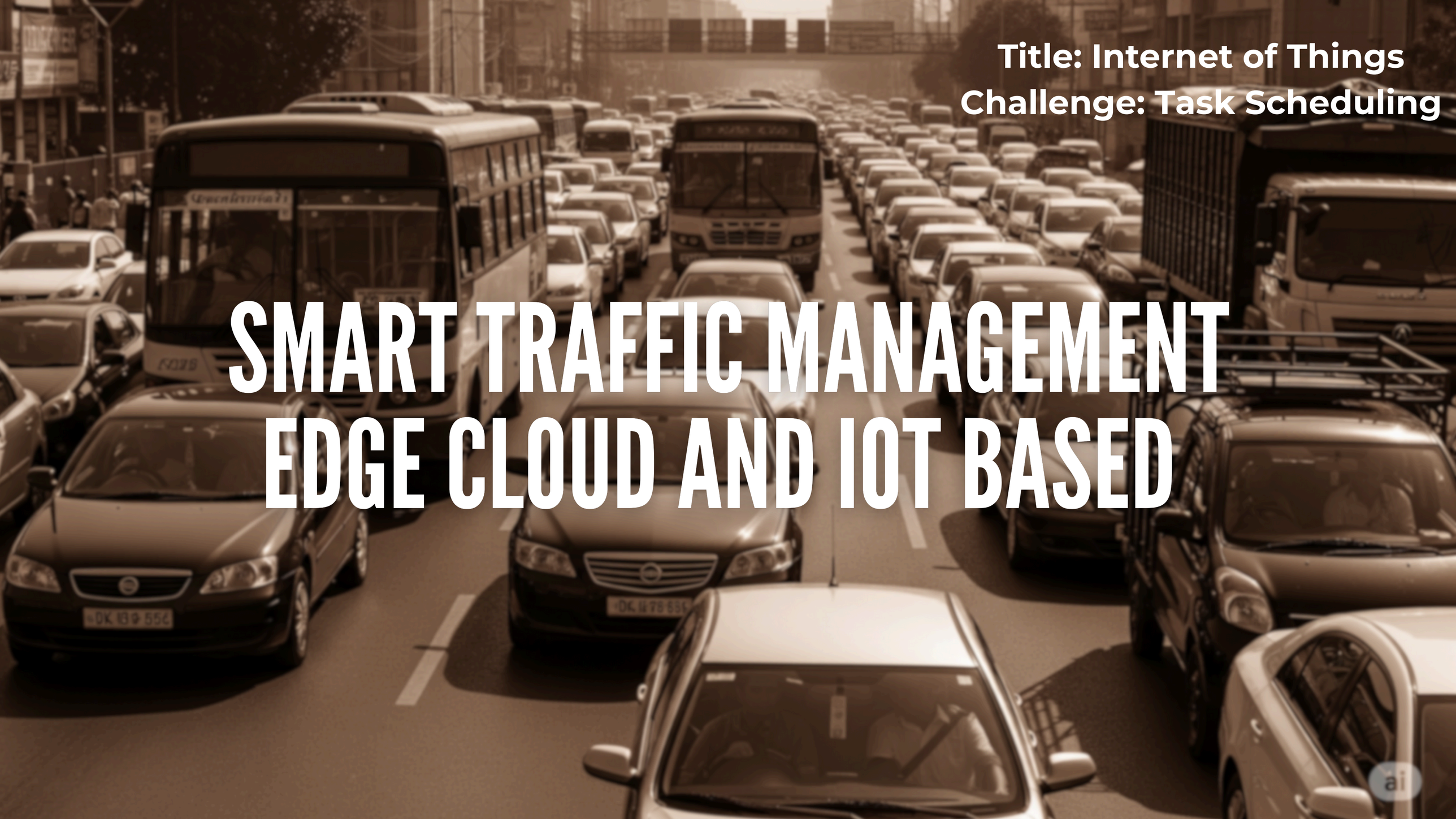


Title: Internet of Things
Challenge: Task Scheduling

SMART TRAFFIC MANAGEMENT EDGE CLOUD AND IOT BASED

Problem Statement

Rapid urbanization, with over 50% of the global population in urban areas and projected to reach 67% in developing countries by 2050, has intensified traffic congestion, resulting in 5.5 billion hours of delays annually, fuel wastage, environmental harm, and increased crash risks at intersections. Traditional traffic signal systems rely on fixed or semi-static schedules, unable to adapt to real-time traffic inflow variations, leading to long queues and inefficiencies. There is a critical need for a task scheduling approach that dynamically adjusts traffic signal phase timings (green, yellow, red) based on predicted vehicle inflow to optimize traffic flow, reduce congestion, and enhance safety at urban intersections.

HOW AND WHY IS IT AN EDGE PROBLEM?

This is an edge problem because real-time decision-making at a local level is critical. Traffic signal control must respond to immediate, localized events (e.g., sudden congestion, pedestrian crossing, emergency vehicles), and centralized cloud systems introduce latency that can be problematic in such time-sensitive scenarios.

Reason	Why Edge is Required
Low Latency	Quick signal changes require sub-second decision-making, only feasible at the edge.
Context Awareness	Edge devices can integrate local events (accidents, road work) into decision logic.
Scalability	Managing many intersections centrally isn't scalable. Edge handles local scaling.
Privacy & Security	Raw video and sensor data stays local, reducing privacy concerns and security risks.
Cost Efficiency	Lower bandwidth costs by avoiding continuous upstream data transfer.

HOW AND WHY IS IT AN EDGE PROBLEM?

Fog acts as a middle layer, handling coordination between nearby intersections and doing heavier processing than the edge, but still faster than the cloud.

WHAT IS THE ROLE OF THE CLOUD?

The cloud manages long-term learning, big data analytics, and system-wide updates — helping improve traffic flow across the whole city over time.

WHY IS EDGE REALLY NEEDED IN THIS PROBLEM ??

Because traffic control needs instant response. Edge devices process data right at the intersection, allowing smart decisions within milliseconds — without relying on internet speed or cloud delays. This ensures faster, safer, and more efficient traffic flow.

IMPLEMENTATION STRATEGY

Core component of Edge Cloud IOT based Smart Traffic Management

Dynamically adjust signal timings (green/yellow/red) in real time

Goal: To reduce congestion and improve road efficiency and road safety

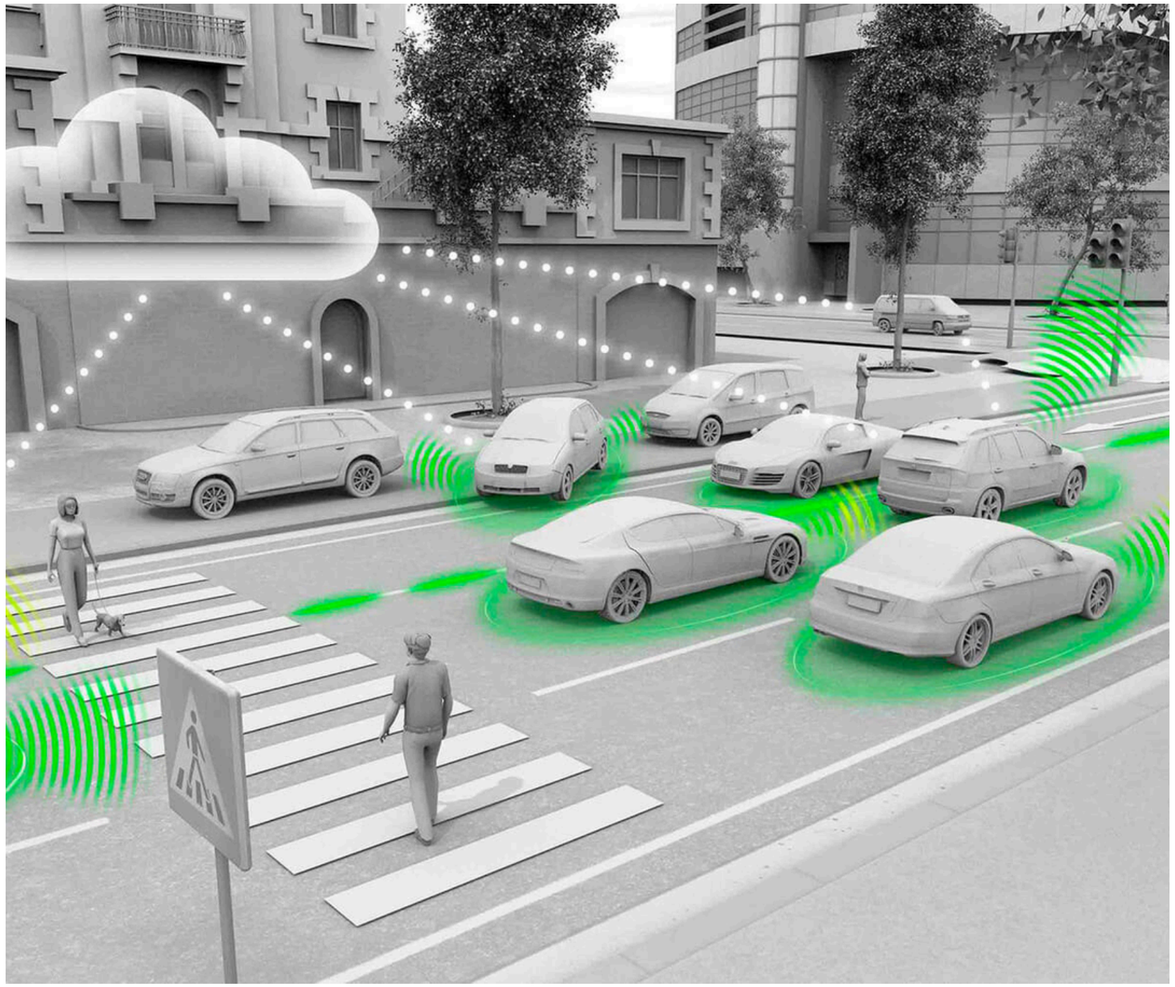
The implementation strategy mainly based on 3 sub systems:

USER SUBSYSTEM

- It handles real time data acquisition
- IOT devices such as **Road Side Units(RSUs) and loop detectors(LDS)**
  Vehicle ID, speed, acceleration and location
- This data is transmitted to cloud using V2I communication
- Ensures **low latency and high reliability**

CLOUD SUBSYSTEM

- Data Manager: Organizes incoming data into
  Mobility
  Traffic flow data
- **Time Calculator** : Computes Time To Reach Signal (TTR)
- Data Analysis :
  - Uses **J48 classifier-Algorithm** 3 (99% accuracy)
  - Predicts if vehicle can pass during signal phase
  - Calculates **Adapted Green Phase Time (AGAP)** based on :
    - Queue length
    - Traffic Speed



Action Sub-System

```
if is_emergency_vehicle_detected():
    send_signal_override("GREEN", lane_id)

elif queue_length > threshold or avg_speed < limit:
    set_signal_time(AGAP)

else:
    maintain_default_cycle()
```

The Action Module operates primarily at the edge level to deliver ultra-low-latency traffic signal control.

It receives inputs like AGAP, vehicle classification, and live traffic data.

Using lightweight decision logic, it adjusts signal timings dynamically and can override signals for emergency vehicles.

This ensures real-time responsiveness, even under network or cloud delays.

HOW ARE WE GOING TO IMPLEMENT ?

STAGE 1: CONGESTION DETECTION (YOLO @ EDGE)

- DATA SOURCE: TRAFFIC CAMERA SENSOR (SIMULATED IN IFOGSIM2, FRAMES FROM SUMO).
- PROCESSING (EDGE):
  - YOLO RUNS ON EDGE DEVICE MODULE.
  - DETECTS VEHICLES, OUTPUTS VEHICLE COUNT + CONGESTION LEVEL.
- OUTPUT TO CLOUD:
  - VEHICLE_COUNT, CONGESTION_LEVEL, TIMESTAMP, LOCATION.

STAGE 2: QUEUE LENGTH PREDICTION (Q-LEARNING @ EDGE)

- DATA SOURCE: YOLO RESULTS + SENSOR METADATA.
- PROCESSING (EDGE):
  - Q-LEARNING AGENT RUNS ON EDGE DEVICE (CLOSE TO SENSOR).
  - PREDICTS QUEUE LENGTH AND SUGGESTS SIGNAL ADJUSTMENTS.
- OUTPUT TO CLOUD:
  - QUEUE_LENGTH, QUEUE_CLASS.

HOW ARE WE GOING TO IMPLEMENT ?

STAGE 3: TRAFFIC CLASSIFICATION (J48 @ CLOUD)

- DATA SOURCE: RESULTS FROM YOLO + Q-LEARNING.
- PROCESSING (CLOUD):
  - J48 DECISION TREE RUNS AS A CLOUD APPLICATION MODULE.
  - USES COMBINED FEATURES (VEHICLE COUNT, CONGESTION, QUEUE LENGTH).
- OUTPUT:
  - TRAFFIC_CONDITION (LOW / MEDIUM / HIGH).

STAGE 4: OUTPUT FORMATTING

- DATA SOURCE: FINAL CLASSIFICATION RESULT.
- PROCESSING (CLOUD): FORMAT STRUCTURED RESULTS.
- OUTPUT: JSON/CSV WITH:
  - TIME OF DAY
  - LOCATION
  - VEHICLE COUNT
  - CONGESTION LEVEL
  - QUEUE LENGTH
  - TRAFFIC CONDITION

STAGE 5: ACTUATION

- ACTUATOR: TRAFFIC LIGHT SIMULATED IN IFOGSIM2.
- PROCESSING: CONTROLLED BY QUEUE PREDICTION + CLOUD CLASSIFICATION.
- OUTPUT: REAL-TIME SIGNAL CONTROL DECISION.

JUSTIFICATION

ULTRA-LOW LATENCY VIA EDGE PROCESSING
REAL-TIME TRAFFIC ACTIONS ARE HANDLED AT THE EDGE TO MINIMIZE DELAYS AND ENSURE IMMEDIATE RESPONSE.

CLOUD FOR INTELLIGENT ANALYTICS
THE CLOUD HANDLES COMPUTE-HEAVY TASKS LIKE AGAP CALCULATION AND J48-BASED CLASSIFICATION USING HISTORICAL DATA.

SEPARATION OF CONCERNS
EDGE PERFORMS EXECUTION AND LOCAL DECISION-MAKING; CLOUD PERFORMS PREDICTION AND PATTERN ANALYSIS.

ADAPTABILITY AND SCALABILITY
CLOUD UPDATES AGAP DYNAMICALLY WITH GLOBAL TRAFFIC INSIGHTS, WHILE EDGE ADAPTS LOCALLY IN REAL-TIME.

RESILIENCE AGAINST NETWORK FLUCTUATIONS
EDGE ENSURES SYSTEM RESPONSIVENESS EVEN DURING NETWORK OR CLOUD DISRUPTIONS.

State of art literature

Using edge computing and AI together has changed traffic control from one big central system to many smart local systems that work in real-time.

These systems focus on fast decisions, analyzing data right where it’s collected, without needing to send everything to the cloud.

Modern traffic systems can now adjust signals automatically and predict traffic jams in advance, helping reduce delays and improve flow.

Year	Approach / Technology	Key Feature	Impact
2020	Edge-Enabled Smart Traffic	Edge nodes for congestion processing	Lower cloud load, faster response
2020	Decentralized RL at Edge	Reinforcement learning at intersections	Adaptive local control
2022	IoT + Digital Twins + Edge AI	Predictive traffic modeling	Proactive routing, less central control
2023	Edge-AI Traffic Monitoring	ML at edge for real-time prediction	Dynamic congestion management
2023	Edge-Based IoT + V2I	Sensors + radar + V2I integration	Instant traffic response

TEAM SPLIT

EDGE/USER MODULE

Simulates IOT devices to collect vehicle data and sending it to edge nodes

BY YUVA HASINI

CLOUD MODULE

Develops edge nodes to run J48 classifier, scheduling green phases using TTR

BY VIRITHA

ACTION MODULE

Simulates dynamic traffic scenarios and triggers system actions based on localized logic.

BY RATAN

TESTING AND IMPLEMENTING

BY VAMSI