# IOT Applications for on Road Safety of Drivers and Vehicles

Thesis Submitted by

**Patrali Pradhan**

## Doctor of Philosophy (Engineering)

Department of Information Technology
Faculty Council of Engineering & Technology
Jadavpur University
Kolkata, India

**2025**

# IOT Applications for on Road Safety of Drivers and Vehicles

*by*

## Patrali Pradhan

Registration Number: 10222090005

*Thesis submitted for the*

## Doctor of Philosophy (Engineering)

Degree of Jadavpur University, Kolkata, India

**Supervisors:**

**Prof. Samiran Chattapadhyay**
Professor (R)
Dept. of Information Technology
Jadavpur University, Salt Lake Campus
Kolkata
West Bengal
India

**Dr. Sudip Misra**
Professor
Dept. of Computer Science and Engineering
Indian Institute of Technology, Kharagpur
Kharagpur
West Bengal
India

**2025**

# Jadavpur University
# Kolkata 700 032, India

**INDEX NO. 102/22/E**

**Title of the thesis :**

IOT Applications for on Road Safety of Drivers and Vehicles

**Name, Designation and Institution of the Supervisors:**

Samiran Chattopadhyay

**Prof. Samiran Chattapadhyay**
Professor(R)
Department of Information Technology
Jadavpur University, Salt Lake Campus
Kolkata-700106
West Bengal
India

Samiran Chottopadhyay
Professor (R)
Deptt. of Information Technology
JADAVPUR UNIVERSITY
Block-LB, Plot-8, Sector-3
Kolkata-700106, India

**Dr. Sudip Misra**
Professor
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur
Kharagpur-721302
West Bengal
India

Sudip Misra
प्राध्यापक /Professor
संगणक विज्ञान एवं अभियांत्रिकी विभाग
Computer Sc. & Engg. Deptt.
[illegible]

# List of Publications

## Journal papers

1. **P. Pradhan**, C. Roy and S. Misra, "Q-Safe: QoS-Aware Pricing Scheme for Provisioning Safety-as-a-Service," in IEEE Transactions on Services Computing, vol. 16, no. 1, pp. 515-524, 1 Jan.-Feb. 2023, doi: 10.1109/TSC.2021.3131658.

Keywords: Pricing; Quality of service; Safety; Roads; Wireless sensor networks; Vehicles; Cloud computing; Road transportation; Service Oriented Architecture (SOA); Decision virtualization; Decision Parameters; Quality of Service (QoS)

2. **P. Pradhan**, C. Roy, S. Misra and S. Chattopadhyay, "Dec-Safe: Dynamic Decision Generation Mechanism for Delivering Safety Services in Vehicular Networks," in IEEE Transactions on Vehicular Technology, vol. 72, no. 12, pp. 15280-15289, Dec. 2023,

doi: 10.1109/TVT.2023.3292210.

Keywords: Safety; Vehicle dynamics; Vehicles; Real-time systems; Delays; Virtualization; Costs; Intelligent transportation systems; Artificial Intelligence; Intelligent Transportation Systems (ITS); Decision virtualization; Safety services; Static decision; Dynamic decision

## International conference papers

1. **P. Pradhan**, C. Roy, S. Misra and S. Chattopadhyay, "Edge Intelligence-Based Safety-as-a-Service Platform for Social IoV Environment," ICC 2023 - IEEE International Conference on Communications, Rome, Italy, 2023, pp. 4943-4948,

doi: 10.1109/ICC45041.2023.10279301. Keywords: Simulation; Training data; Bandwidth; Reinforcement learning; Probability density function; Delays; Safety-as-a-Service; Social IoV (SIoV); Edge In-telligence; Ultra-low latency; Decision Virtualization; Artificial Neural Network (ANN)

2. **P. Pradhan**, Samiran Chattopadhyay, Subrata Chowdhury, 2024. "Risk Assessment and Safety-as-a-Service Provisioning in IoV Networks: A Roadmap". In Proc. 5th IEEE Global Conference for Advancement in Technology (GCAT), Karnataka, India. Oct 4-6, 2024.

## Others

1. **Patrali Pradhan**, Chandana Roy, Sudip Misra, Samiran Chattopadhyay, "Safe-Price: Dynamic Pricing Mechanism For Provisioning Safety Services in IoT Environment",2024. Submitted in the IEEE Transactions on Intelligent Transportation Systems.

## List of Presentations in National/International/ Conferences/ Workshops:

1. **P. Pradhan**, C. Roy, S. Misra and S. Chattopadhyay, "Edge Intelligence-Based Safety-as-a-Service Platform for Social IoV Environment," ICC 2023 - IEEE International Conference on Communications, Rome, Italy, 2023, pp. 4943-4948,
   doi: 10.1109/ICC45041.2023.10279301.
   Keywords: Simulation; Training data; Bandwidth; Reinforcement learning; Probability density function; Delays; Safety-as-a-Service; Social IoV (SloV); Edge In-telligence; Ultra-low latency; Decision Virtualization; Artificial Neural Network (ANN)
2. **P. Pradhan**, Samiran Chattopadhyay, Subrata Chowdhury, 2024. "Risk Assessment and Safety-as-a-Service Provisioning in IoV Networks: A Roadmap". In Proc. 5th IEEE Global Conference for Advancement in Technology (GCAT), Karnataka, India. Oct 4-6, 2024.

Samiran Chattopadhyay

PROFESSOR
Deptt. of Information Technology
JADAVPUR UNIVERSITY
Block -LB, Plot-8, Sector-3
Salt Lake, Kolkata-700106, India

Sudip Misra
प्राध्यापक / Professor
संगणक विज्ञान एवं अभियांत्रिकी विभाग
Computer Sc. & Engg. Deptt.
भा प्रौ सं खड़गपुर/IIT Khargapur

PROFORMA – 1

## Statement of Originality

I, **Ms. Patrali Pradhan**, registered on **12/05/2022** do hereby declare that this thesis entitled "**IOT Applications for on Road Safety of Drivers and Vehicles** " contains literature survey and original research work done by the undersigned candidate as part of Doctoral studies.

All information in this thesis have been obtained and presented in accordance with existing academic rules and ethical conduct. I declare that, as required by these rules and conduct, I have fully cited and referred all materials and results that are not original to this work.

I also declare that I have checked this thesis as per the "Policy on Anti Plagiarism, Jadavpur University, 2019", and the level of similarity as checked by iThenticate software is **8%**.

Signature of Candidate:

Patrali Pradhan

(Patrali Pradhan)
Date : 28/03/25

Certified by Supervisors:
(Signature with date, seal)

1. Samiran Chattopadhyay
(Samiran Chattapadhyay)

Samiran Chottopadhyay
Professor
Deptt. of Information Technology
JADAVPUR UNIVERSITY
Block-LB, Plot-8, Sector-3
Salt Lake, Kolkata-700106, India

2. Sudip M
(Sudip Misra)

Sudip Misra
प्राध्यापक / Professor
संगणक विज्ञान एवं अभियांत्रिकी विभाग
Computer Sc. & Engg. Deptt.
भा प्रौ सं खड़गपुर/IIT Khargapur

PROFORMA – 2

# CERTIFICATE FROM THE SUPERVISORS

This is to certify that the thesis entitled "**IOT Applications for on Road Safety of Drivers and Vehicles** " submitted by **Ms. Patrali Pradhan**, who got her name registered on **12/05/2022** for the award of Ph.D. (Engg.) degree of Jadavpur University is absolutely based upon her own work under the supervision of **Prof. Samiran Chattapadhyay, Professor (R), Department of Information Technology, Jadavpur University, Salt Lake Campus, Kolkata, India** and **Prof. Sudip Misra, Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur, India** and that neither her thesis nor any part of the thesis has been submitted for any degree/diploma or any other academic award anywhere before.

Signatures of the Supervisors with Date and Official Seal

Samiran Chattopadhyay

**Prof. Samiran Chattopadhyay**
Professor (R)
Department of Information Technology
Jadavpur University, **Salt Lake Campus**
Kolkata 700106
West Bengal, India

Samiran Chottopadhyay
Professor
Deptt. of Information Technology
JADAVPUR UNIVERSITY
Block-LB, Plot-8, Sector-3
Kolkata-700106, India

Sudip Misra

**Dr. Sudip Misra**
Professor
Department of Computer Science and Engineering
Indian Institute of Technology
Kharagpur 721302
West Bengal, India

Sudip Misra
प्राध्यापक /Professor
संगणक विज्ञान एवं अभियांत्रिकी विभाग
Computer Sc. & Engg. Deptt.

# Acknowledgement

I would like to express my sincere gratitude to everyone who supported me throughout the journey of completing this thesis. This work would not have been possible without their guidance, encouragement, and assistance.

First and foremost, I am deeply grateful to my supervisor, Prof. Samiran Chattopadhyay, for his invaluable support, guidance, and expertise. His constructive feedback and encouragement has been instrumental in helping me complete this research. I am especially thankful for the countless hours they dedicated to reviewing drafts and discussing ideas with me, helping me academically and personally grow.

I am also deeply grateful to my Co-Supervisor, Prof. Sudip Misra, for his unwavering support and encouragement during one of the most challenging times in my life. Amid personal difficulties, his guidance and motivation were instrumental in helping me persevere. His invaluable insights and thoughtful feedback greatly enriched the quality of this thesis.

A special thanks goes to Dr. Chandana Roy, whose support and companionship were invaluable throughout this journey. Her friendship, insightful discussions, and feedback were instrumental in helping me refine my ideas and approaches.

Finally, I am deeply grateful to my family for their unconditional love, patience, and encouragement. Their support has been my foundation; without them, none of this would have been possible. Also, I want to express my heartfelt gratitude to all SWAN LAB (IIT, KGP) members, all the professors especially Dr. Tohida Rahman, and staffs of the Department of Information Technology, JU, Kolkata.

Thank you all for believing in me and for helping me to make this work a reality.

Patrali Pradhan

*To*

*my Husband (Late Jayanta Sen), my Son (Anwit Sen), Parents and Siblings*

*"Optimism is the faith that leads to achievement. Nothing can be done without hope and confidence."*

– Helen Keller

# Abstract

In the past few years, the integration of automation techniques with the traditional transportation system has minimized on-road hazards and traffic congestion. Further, this also assists in communication among the vehicles. The researchers have developed diverse techniques and methods such as Advanced Driving Assistance System (ADAS), Intelligent Transportation System (ITS). However, none of these techniques provide prior safety-related information to the users. Safety-as-a-Service (Safe-aaS) is one of the uniquely developed platforms that provide safety-related customized information to the users.

Typically, Safe-aaS comprises five layers- device layer, edge layer, decision layer, decision virtualization layer, and application layer. In the Safe-aaS environment, heterogeneous forms of stationary and mobile sensor nodes are present in the device layer. The stationary sensor nodes are deployed at a particular geographical location. On the other hand, sensor nodes are either inbuilt or externally placed into the vehicles. These sensor nodes sense and transmit data to the edge layer/cloud, depending upon the time-critical nature of data. Thereafter, the primarily processed sensed data is transmitted to the decision layer for further processing. The generated decisions are logically mapped with the user's selected decision parameters in the decision virtualization layer. Finally, the customized decisions are provided to the users. During registration, the users provide their credentials, select the decision parameters, and make payment through the web portal. In a road transportation environment, decision parameters may include a number of sharp turnings, number of potholes, speed limit, distance between neighboring vehicles, sudden weather conditions, and driver's behavior.

In a Safe-aaS implemented scenario, a huge volume of data is generated and transmitted to the cloud/edge. Therefore, the processing, analysis, and storage management of this colossal volume of data is a complex task. In case of any delay in delivery of their decision/delivery of incorrect decisions, may lead to a hazardous situation. On the other hand, the sensor nodes present in the device layer, are energy-constraint in nature. Sensing, processing, and computation of the same data may result in unnecessary energy consumption. Therefore, it is necessary to conserve this energy for future applications as well as timely delivery of accurate decisions.

To fulfill the above-mentioned criteria, we propose the adaptive decision-generation mechanism named as "Dec-Safe". In " Dec-Safe", we categorize decisions as static and dynamic. Static decisions are generated from the relationship mapping between static parameters such as a number of sharp turns on a road, the location of potholes, and the location of speed breakers, whose value does not vary frequently with time as static. Therefore, the utilization of storage units and the number of overall sensor nodes required are reduced. As a result, the overall energy utilized for decision generation is reduced. On the other hand, dynamic decisions are those whose value varies frequently such as weather, road conditions, and sudden hazardous situations. Dynamic decisions are generated when users select dynamic decision parameters. The decision parameters selected by the users may overlap with each other. We apply the clustering method to extract similar decision parameters. Therefore, the same dynamic decision is delivered to multiple users simultaneously with minimum delay and certain customization.

In Safe-aaS, the registered users access the safety services during their journey from the source to the destination. The users make payments for the services through the web portal.Safety Service Provider (SSP)s are the centralized entities, which manage the entire Safe-aaS platform. The sensors and vehicle owners rent their sensor nodes to the Safe-aaS platform and receive the amount. The profit of the SSP is the remaining amount after providing the rent to the sensor and vehicle owners, and maintenance charges. Therefore, complex transactions take place among these sensor and vehicle owners, SSPs, and users. As these decisions are time-critical, maintaining the Quality of Service (QoS) and providing optimal prices to the users for their requested decision parameters, are major concerns of the SSPs. Based on these above-mentioned reasons, the SSP suggests low and high-price parameters to the users periodically.

Typically, in Safe-aaS, customized time-critical decisions are generated and delivered to the users. The processing, analysis, and storage of this generated data is quite complicated. Moreover, another major challenge associated with the Safe-aaS platform is to provide accuracy in the generated decisions. To improve the accuracy and minimize the latency incurred in decision generation, we place the edge servers at the network edge. Therefore, unlike the traditional Safe-aaS platform, the analysis and storage of data are done at the user's location. The edge servers are deployed at certain geographical locations. As a result, the overall computational and processing costs are also minimized. Therefore, we introduce the edge intelligence layer in the Safe-aaS platform to provide accurate decisions with ultra-low latency. We apply one of the popular Artificial Intelligence (AI) models, Artificial Neural Network (ANN), at the edge nodes to classify and select edge servers. The fuzzified decisions are generated at the edge server, and further, these decisions are propagated to the decision layer. Therefore, latency in decision delivery is also minimized.

In the Safe-aaS environment, the users are likely to select decision parameters as per their requirements. However, we introduce that based on their geographical location, the decision

parameters are suggested by the SSP. In that scenario, the users are not charged for enjoying the safety services. Generally, the processing cost of the decision parameters is quite high. Therefore, the total price charged from the users is based on the selection or de-selection of parameters or number of times safety services are availed by them. In such cases, the decision is not generated by the ANN on the server side if there is no parameter selection. Hence, the profit of the SSP is determined by estimating the cash outflow and inflow of the number of users, sensor types, number of active sensors, and their geographical location.

**Keywords— IoT, IIoT, IoV, ITS, ADAS, Autonomous Vehicles, Safe-aaS, Road-Safety, Safe-Driving, Service-Oriented Architecture, RSU, Sensors, AI, ANN, Machine Learning**

# Contents

# List of Figures

# List of Tables

# List of Algorithms

# List of Acronyms

| | | |
|---|---|---|
| **AI** | Artificial Intelligence | |
| **IoT** | Internet of Things | 1 |
| **IIoT** | Industrial Internet of Things | 1 |
| **IoV** | Internet of Vehicles | 1 |
| **SIoV** | Social IoV | 16 |
| **ITS** | Intelligent Transportation System | |
| **ADAS** | Advanced Driving Assistance System | |
| **VANET** | Vehicular Adhoc Network | 15 |
| **V2V** | Vehicle-to-Vehicle Communication | 1 |
| **V2I** | Vehicle-to-Infrastructure communication | 1 |
| **V2X** | Vehicle-to-Everything | 19 |
| **AoI** | Angle of Inclination | 14 |
| **ANN** | Artificial Neural Network | |
| **QoS** | Quality of Service | |
| **SOA** | Service-Oriented Architecture | 2 |
| **DSRC** | Dedicated Short Range communications | 16 |
| **RSU** | Road-Side Unit | 1 |
| **ML** | Machine Learning | 1 |
| **RL** | Reinforcement Learning | 17 |
| **RF** | Random Forest | 17 |
| **RNN** | Recurrent Neural Network | 17 |
| **LSTM** | Long Short Term Memory | 17 |
| **BP** | Backpropagation Neural Network | 18 |
| **MLS** | Mobile Laser Scanning | 14 |

| | | |
|---|---|---|
| **Safe-aaS** | Safety-as-a-Service | |
| **SSP** | Safety Service Provider | |
| **GPS** | Global Positioning System | 16 |
| **LiDAR** | Light Detection and Ranging | 18 |
| **SVM** | Support Vector Machine | 17 |
| **D2ITS** | Data Driven Intelligent Transportation System | 16 |
| **DDC** | Drowsiness Detection Classifier | 15 |
| **CDF** | Cumulative Distribution Function | 98 |

# 1

# Introduction

In the last few decades, applications of wireless and mobile communication technologies have been used to implement the Internet of Things (IoT) [1] in the transportation industry. The automation of vehicles and subsequent improvement in on-road vehicle safety [2] has been driven by IoT-based technology in the transportation sector. With the additional diverse static and mobile sensors, these on-road vehicles gained further intelligence and security. Mobile sensor nodes are incorporated or deployed within vehicles, whereas static sensor nodes are placed at various locations. In Industrial Internet of Things (IIoT) [3], [4], [5], various physical devices, sensors, and machines are interconnected and communicated through the Internet, and IoT-based solutions [6] help to address various industrial problems. Internet of Vehicles (IoV) [7], an essential domain of the Industrial Internet of Things (IIoT), improves the traffic control system [8], [9] to handle vehicular traffic smoothly by reducing on-road accidents and traffic congestion in the transportation industries. In the IoV environment, vehicles equipped with heterogeneous sensor nodes and Road-Side Unit (RSU)s communicate among themselves through mobile or wireless networks. IoV has proved to be the most remarkable application of IoT [10] in the field of on-road automobiles. Modern vehicular networks mainly focus on Vehicle-to-Vehicle Communication (V2V), and Vehicle-to-Infrastructure communication (V2I) communications. Tang *et al.* [11] focused on challenges that arise in a real-time environment with a 6G network. They identify resource allocation and network traffic delay as the significant challenges and focus on the solution approaches by applying intelligent tools such as Machine Learning (ML). ML-based intelligent systems are capable of solving several issues in vehicular networks [12–14].

On-road safety or driving safety is the major concern in the growing number of vehicles. In addition, a large amount of data is generated with the increasing rate of autonomous vehicles. To ensure provisioning safety, existing research works have tried to solve various on-road safety concerns, such as driver drowsiness [15], [16], detection of manhole covers [6], trajectory planning, and other safety threats. Therefore, IoV improves traffic management by implementing the ITS [2], ADAS [17], to minimize the rate of accidents and congestion. Cutting-edge technologies such as ANN, and ML, are utilized in intelligent IoV [18] to detect and predict traffic congestion. Researchers in this article use ML models to classify on-road traffic based on the situation. Analyze the reasons for traffic congestion in smart cities, examine its effects, and propose solutions.

In the context of road transportation, prior information regarding real-time safety helps to avoid hazardous conditions in dynamic environments. Safe-aaS [19] is a newly developed cloud-based platform where customized safety-related decisions are provided as a service to the end users. It provides safety-related information on a pay-per-use basis. The users register for the service through the web portal/application. After successful login, they select certain decision parameters such as inter-vehicular distance, number of turnings and potholes, road conditions, and weather conditions at the time of registration. Based on the selection, customized decisions are provided to the users.

## 1.1 Safe-aaS: Overview and Challenges

Safe-aaS is a Service-Oriented Architecture (SOA) with five layers such as the Device, Edge, Decision, Decision Virtualization, and Application layer as in Fig. 1.1. The device layer comprises heterogeneous types of static and mobile sensor nodes, which sense and transmit their sensed data to the edge/cloud. The static sensor nodes are installed at a particular

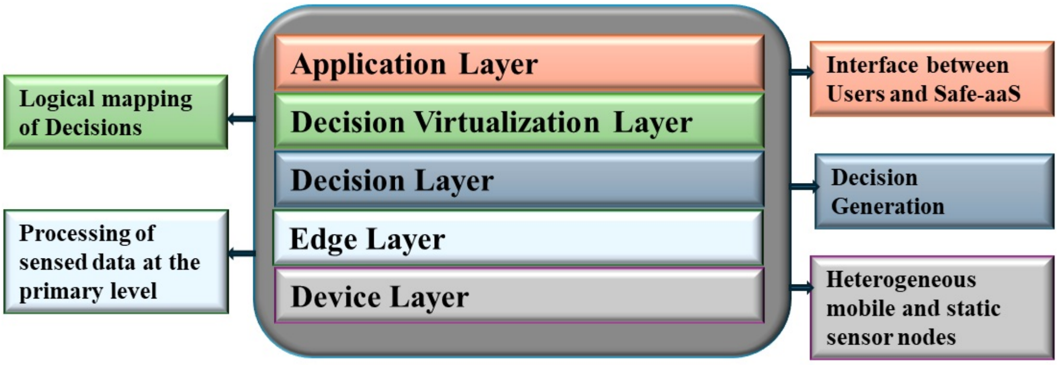


FIGURE 1.1: Layers of Safe-aaS Architecture

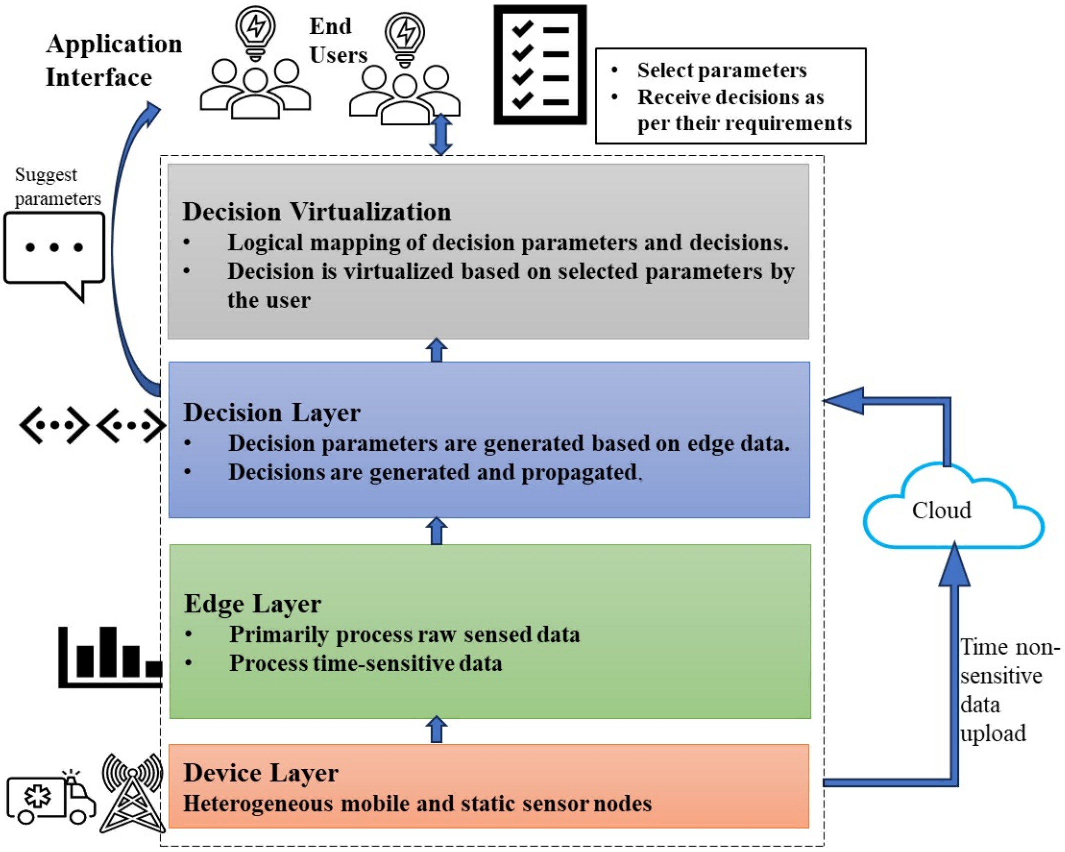


FIGURE 1.2: Layer Wise Processes

geographical location, while vehicles are equipped with mobile nodes. Based on their time-sensitive nature, these sensed data are primarily processed at the edge nodes. In addition, these primarily processed data are delivered to the decision layer, where the decision is generated. Finally, the logical alignment of decision parameters specified by end-users and the resulting decisions is accomplished within the decision virtualization layer. Conversely, the application layer serves as the intermediary between end-users and the Safe-aaS infrastructure.

Layer-wise processes of Safe-aaS are described in Fig. 1.2. End-users register, select certain decision parameters, and make payments through the Web portal. The safety services are provided to the users in the form of safety decisions, depending upon their requirements. In a dynamic environment, the decision parameters may be the number of potholes, sharp turnings, traffic signals, weather conditions, and causes of congestion/accidents.

In a Safe-aaS environment, similar decisions are delivered to multiple users. Therefore, re-sensing, transmitting, processing, and storing the decisions incur unnecessary energy consumption and time delay. Moreover, the sensor nodes are energy-limited in nature. Therefore, it is necessary to conserve their energy for future utilization. Generated decisions are time-sensitive, so, any delay in the delivery of decisions may cause sudden accidents or other hazardous situations. The edge nodes are not capable of processing time-sensitive information beyond their capacity with limited storage space.

From a business perspective, in Safe-aaS, various actors such as sensor vendors, vehicle owners, end users, and SSPs play major roles.The sensor and vehicle owners lease their sensor nodes and receive compensation as determined by the Service Supply Providers (SSPs). In addition, end-users benefit from these safety services on a pay-per-use basis. Therefore, the remaining amount from the payment done by the end-users and the rent paid by the sensor and vehicle owners is the profit of the SSP. The SSPs possess a tendency to maximize their profit, while the end-users desire to avail of these services at a lower price. However, as safety-related decisions are delivered to the end users, it is essential to maintain the quality of the safety-related decisions provided to them.

## 1.2 Key Challenges in Safe-aaS

- **Resource Optimization:** To manage the large number of sensor nodes, storage space, and minimizing response time delays, is a highly complex task during decision generation. Re-sensing, processing, and storing massive volumes of data repeatedly leads to unnecessary resource consumption.
- **Pricing Mechanism:** To design an effective pricing model that balances the SSPs and users while maintaining high QoS.
- **Processing Overload at Cloud Layer:** Continuous data transmission from both mobile and static sensor nodes leads to a heavy processing burden on cloud servers, resulting in delays and inefficiencies in real-time decision-making.
- **Decision Flexibility and Efficiency:** To develop a mechanism that adapts to both static and dynamic decision parameters, while ensuring timely, accurate, and cost-efficient delivery of decisions to users.

## 1.3 Motivation and Scope of The Work

In transportation industries, safety is a major concern with the increasing number of on-road vehicles. Prior delivery of safety-related information such as road conditions, weather, maneuver detection, and frequency of road accidents, reduces the risks of congestion or road accidents. We consider a Safe-aaS platform for delivering tailored safety-related recommendations to users. The safety-related decisions are generated based on selected decision variables by the end users after login to the web portal. Typically, decision variables are location, and number of potholes, number of turns. The static decision parameters, generated from roadside sensors placed in various geographical locations, are not modified frequently. On the other hand, the mobile sensor nodes are attached to the vehicles, and sense the dynamic environment such as weather conditions, road conditions, and congestion.

Safe-aaS infrastructure comprises heterogeneous mobile and static sensor nodes that sense and transmit data to the edge/cloud for primary processing. These edge nodes are placed in various geographical locations with limited storage and computation capability. In real-time situations, time-critical data is partially processed at the edge nodes. Additionally, in a dynamic environment, time-critical decisions are delivered with a certain time delay which may lead to sudden road accidents or hazardous conditions. Consequently, sensor nodes collect and transmit analogous data to the nearest edge nodes or cloud, depending on the time sensitivity of the information. Thus, in such situations, re-sensing, processing, and computation on the same data to generate decisions, results in unnecessary energy and time consumption. On the other hand, these sensor nodes, placed on roadsides, are energy constrained in nature. In the above scenario, any middle-layer processing entities such as edge servers lower the storage and computing burden of the edge/cloud. In existing research, it is proved that several AI models/methods can reduce the computation overhead of any entity in the network and lower the time delay for delivering messages in IoV. By optimizing sensor usage, unnecessary energy consumption can be minimized.

From the business perspective, in Safe-aaS, end users register themselves for the safety services through the web portal, request decision parameters, and make payments for service usage. Safety-related decisions are generated based on the selected decision parameters. Users select parameters from the parameters suggested by SSP. There is a possibility of rejection to choose the parameters by the users at any moment during their journey. In addition, the parameters suggested by the SSP, are not suitable for generating decisions in the current situation. Finally, users make payments for the overall service usage through the web portal.

The satisfaction of the end-users for the QoS provided by the SSP is another important issue in Safe-aaS. On the other hand, multiple actors such as sensor owners, vehicle owners, end-users, and SSPs, exist in the Safe-aaS infrastructure. Therefore, complex cash inflow and outflow take place for each of the actors, which infuses a business perspective. Further, the existing pricing schemes do not provide customized prices for decisions, as per the varying demand of each end-user. In the Safe-aaS architecture, decision virtualization results in dynamic changes in the price charged by the SSP. As no specific pricing scheme exists for the use of Safe-aaS, we design a dynamic and differential pricing scheme to use with it.

Existing research works on Safe-aaS focus on minimizing response time delay [20], energy consumption [21], optimizing total sensor nodes usage [22], and pricing [23]. We identify certain research lacunae related to resource management, as illustrated in Fig. 1.3.

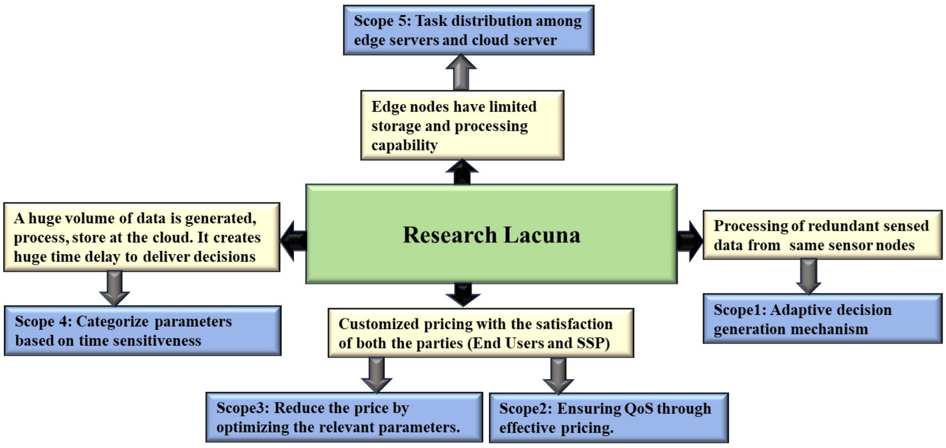


FIGURE 1.3: Scope of the Thesis

In particular, we consider the application scenario of Safe-aaS in road transportation and focus on the management of resources such as storage capacity, total number of sensor nodes usage, effective energy consumption, response time delay, and QoS-aware pricing.

**The scope of the Thesis is outlined as follows:**

- In Safe-aaS, resources like sensor data storage, energy for data processing, and timely decision-making are critical for ensuring efficient and effective safety services. This objective is to create a system that continuously adapts based on real-time conditions to ensure the efficient and prompt delivery of safety-related information to the end users. We Propose a mechanism that dynamically adjusts decisions to optimize the use of key resources such as storage, energy consumption, and sensor deployment, and minimize response time delays. The mechanism would intelligently optimize resources depending on current demand and system load, ensuring a balance between cost-effectiveness and performance.

- QoS plays a significant role in customer experience, especially for safety services where timely and reliable responses are crucial. This objective aims to create a pricing mechanism that adjusts based on QoS metrics such as latency, availability, and reliability. In Safe-aaS, users pay according to the service use as per their requirements. Service providers benefit from the mechanism that allows them to scale their service offerings while ensuring profitability. This mechanism ensures the needs of both users (who want high performance at low costs) and service providers (who need to maintain profitability and service quality). We develop a pricing strategy that ensures both user satisfaction and service satisfaction for SSPs.

- In Safe-aaS, safety-related decisions are generated and delivered from the centralized server. The prompt decision delivery to the end users is crucial for on-road safety.

The introduction of the edge intelligence layer may ensure decentralized computation and decision-making by leveraging local devices (such as edge servers) closer to where the data is generated from sensors or IoT devices. This reduces the processing burden on central servers, minimizes latency, and ensures that real-time decisions are made locally, enhancing system responsiveness and reducing bandwidth usage. The edge layer performs tasks like pre-processing, data filtering, and local analytics, allowing for faster, real-time decisions, particularly for time-sensitive safety applications.

- Safety services often require instant responses, especially in critical situations such as emergency alerts or hazard detection. This objective focuses on creating a pricing model where users can pay for prioritized access to safety decisions with ultra-low latency. The mechanism would take into account various factors such as the level of urgency, system load, and available resources, offering premium services for ultra-fast decision-making while ensuring that this pricing structure is still cost-effective for users. We design a pricing mechanism that guarantees users can receive safety-related decisions with minimal delay.

- The dynamic environment refers to the fluctuating nature of risks, threats, and hazards due to factors such as weather changes, and infrastructure malfunctions. SSPs must continuously monitor these changes in real-time, using sensors, analytics, and data streams, to provide up-to-date safety recommendations. A dynamic environment requires SSPs to make quick decisions, such as advising evacuations, shutting down critical infrastructure, or altering traffic routes to prevent accidents. SSPs function within specific geographic areas, such as urban centers, industrial facilities, or smart cities, to monitor environmental conditions, security threats, and safety measures. Safety recommendations by the SSPs within their operating zone after analyzing safety SSPs must continuously monitor these changes in real-time, using sensors, analytics, and data streams, to provide up-to-date safety recommendations. A dynamic environment requires SSPs to make quick decisions, such as advising evacuations, shutting down critical infrastructure, or altering traffic routes to prevent accidents.

## 1.4 Problem Statement and Objective

In this Thesis, we provide the technological solutions to manage resources such as the total number of sensor nodes, required storage space, and response time delay during decision generation. Also, we concentrate on the effective pricing scheme in Safe-aaS infrastructure.

We explore the transportation sector industries as the application area of Safe-aaS infrastructure. In Safe-aaS, after successfully logging into the web portal, customized safety-related decisions are generated and delivered to the end users based on their selected decision parameters. Decision virtualization is introduced in Safe-aaS, which provides the same decision to multiple users dynamically. Moreover, end users select a certain number and type of decision parameters as per their requirements. Static decisions should be generated every time when users select static decision parameters that are not modified frequently and dynamic decisions must be delivered to the users whenever any modification in dynamic parameters such as weather, certain road conditions, or driver's behavior. We propose an adaptive decision-generation mechanism based on static and dynamic decision parameters to minimize storage, computation complexity, and time delay. With this proposed mechanism, flexibility to switch any of the modes- static and dynamic at any time.

Heterogeneous static and mobile sensor nodes in the device layer of the Safe-as-a-Service infrastructure are responsible for sensing and transmitting huge volumes of data to the edge/ cloud depending upon the time-sensitivity. Similar data is generated and transmitted to the edge/cloud from the roadside sensors. Therefore, re-sensing, processing, and storing this colossal amount of data is a complex task. Task distribution to the edge servers, placed near to the edge nodes minimizes computation and processing delay. These edge servers have the capability of storing and processing time-sensitive data. Therefore, overall time delay and task overload are minimized in the cloud. We introduce an edge-intelligence layer in traditional Safe-aaS infrastructure.

In Safe-aaS, various key users such as end users, SSPs, sensor owners, and vehicle owners play a key role. End users register themselves into the web portal to receive safety-related decisions and provide credentials including source and destination location for their journey. Finally, in the end, they make payment for the service usage through a web portal. The SSPs pay rent to the sensor owners for the sensors deployed in the vehicles. There are certain cash outflow and inflow present in this scenario. Therefore, optimization of sensor usage helps to reduce a certain amount of cash outflow from the SSP side. We can reduce sensor usage for static decision generation. Sensors present in vehicles and at the roadside, generate the same parameters. Therefore, re-sensing is not required if there is a logical mapping between static parameters and static decisions. And these parameters are treated as low-price parameters and dynamic parameters are typically considered as high-price parameters. Simultaneously, QoS is a major concern in Safe-aaS. QoS is measured based on the optimum sensor usage, data processing cost, and overall response time delay.

The specific objectives of this Thesis are as follows:

- **Adaptive Decision Generation Mechanism:** Development of an adaptive mechanism for generating decisions that optimize key resources such as storage space, energy consumption, sensor usage, and response time. By classifying decision parameters as either static or dynamic, the system can avoid re-sensing and re-processing data for static parameters that do not change frequently. This reduces the overall load on sensors, storage, and computational resources. For dynamic decisions, the system can process updates only when there are changes in dynamic parameters (such as weather, traffic, or driver behavior), ensuring real-time responsiveness. The proposed adaptive mechanism provides flexibility to switch between static and dynamic decision modes, ensuring efficient use of resources.

- **QoS-Aware Pricing Mechanism:** The thesis aims to develop a pricing model that satisfies both the end users and service providers (SSPs). Static decisions, which rely on parameters that change infrequently, are considered low-cost, while dynamic decisions, which require real-time data processing, are priced higher due to the greater computational and sensor usage demands. The pricing model takes into account with some factors such as sensor usage optimization, data processing costs, and overall response time delay, balancing cost efficiency with service quality for both SSPs and end users. By reducing unnecessary sensor activations for static decisions, the model minimizes the cash outflow for service providers, while still ensuring timely and accurate decision delivery to users.

- **Introduction of The Edge-Intelligence Layer:** To address the issue of processing overload at the cloud level, the thesis introduces an edge intelligence layer in the Safe-aaS architecture. Edge servers, located closer to the data-generating sources (such as roadside, and vehicle sensors), are used to process and store time-sensitive data. This reduces the computational and processing burden on the cloud, resulting in lower latency and faster decision generation. By offloading tasks to the edge, the infrastructure can handle large volumes of sensor data more efficiently, ensuring that decisions are delivered to users with minimal delay.

- **Ultra-Low Delay-Based Pricing Mechanism:** The thesis proposes a pricing mechanism that guarantees ultra-low delay in delivering safety-related decisions to users. This mechanism would incentivize faster processing and prioritize time-sensitive decisions, particularly for dynamic safety parameters. The pricing model ensures that users receive safety decisions with minimal delay, which is critical in scenarios like real-time traffic updates, weather alerts, or accident avoidance recommendations. Task distribution between edge servers and the cloud is optimized to minimize the time lag in decision generation and delivery.

- **Development of Safety Recommender System:**
  The thesis also aims to develop a safety recommender system that operates using Safe-aaS platform services. This system would provide customized safety recommendations based on users' journey data (such as source and destination), preferences, and real-time

sensor inputs. The recommender system would generate both static and dynamic safety decisions, depending on the user's needs, providing personalized safety guidance during their travels. By leveraging both static and dynamic parameters, the recommender system ensures that users receive relevant and timely safety information, enhancing overall road safety and user satisfaction.

## 1.5 Contributions

Considering the objectives mentioned above, the key contributions of this thesis are outlined below.

- We propose an adaptive decision generation mechanism, **Dec-Safe [24]**, which is a robust and flexible scheme for the end users. In this mechanism, based on the selection of decision parameters (static and dynamic), the decision generation (static/ dynamic) method is altered for the users as per their requirements during their whole journey. In the static approach, the users select decision variables, whose values do not fluctuate frequently with time. However, they may add some additional parameters at any time, based on real-time requirements. In the case of a dynamic approach, the type and number of parameters requested by the users may vary from their starting to destination point. In our scheme, users can switch their service mode anytime, anywhere. Because of this flexibility, users can get cost-effective good quality service minimizing storage space in the cloud, effective energy consumption, and sensor usage.
- We design a suggestion-based pricing scheme, **Q-Safe [25]**, where the SSPs provide decision parameters to end-users based on their geographical position. We categorize the decision parameters selected by the end-users as low-cost and high-cost variables. Further, we estimate the effective total cost from the optimal number of selected decision parameters. In Safe-aaS, the end-users possess a tendency to avail of the safety services at a lower price, while the SSPs desire to enjoy higher profits. To satisfy both the SSPs and the end-users, we formulate the total cost as an optimization function.
- We present the concept of an edge intelligence layer in the Safe-aaS platform [26], which comprises distributed edge servers. we compute total delay with/without the edge intelligence layer. We observe that with the introduction of the edge intelligence layer, the total delay is minimized compared to the delay without the edge layer.
- We introduce a scheme to jointly minimize the latency incurred in decision generation and the payment received from the users. To increase the usage of the safety services, SSP suggests active decision variables to the users. These parameters are produced from

the corresponding activated sensor nodes. On the other hand, the registered users chose these decision variables to receive safety-associated decisions.

- We propose a decision recommendation system that takes into account various factors such as weather, road conditions, and the driver's risk profile. This system will analyze safety-related decision parameters to calculate a safety score and a risk score for each decision parameter. The proposed system not only generates a personalized safety plan for the user but also shares this plan with neighboring vehicles, fostering a collaborative approach to road safety. This integration of multiple safety factors into a cohesive recommendation system represents a novel contribution to the field, aiming to enhance the overall safety of road transportation environments.

In summary, the thesis addresses:

- Resource optimization (sensor usage, storage, time).
- Pricing mechanisms to balance cost and service quality.
- Task distribution and edge-intelligence to handle processing load.
- Flexibility in decision-making (static vs. dynamic).
- A safety recommender system for real-time, customized user guidance.

## 1.6 Organization of the Thesis

In this section, the organization and content of the following chapters are described briefly.

**Chapter 2** provides a thorough analysis of the research literature on vehicular safety, driving safety, and road safety for on-road vehicles to provide Safety-as-a-Service to the transportation industry. A comprehensive literature survey reveals several on-road driving risk factor sets and addresses safety solutions for those. It explains methods for technical solutions using several mathematical models, including categorization models. Therefore, we identify the limitations and future scopes and explore the approaches, methodology, technologies, and modalities offered in such works.

**Chapter 3** presents *Dec-Safe*, an adaptive decision generation mechanism where we present two approaches- static and dynamic. The safety services are provided to the users in the form of safety decisions, depending upon their requirements. Several environment variables are considered decision variables or parameters in this mechanism. End users have the flexibility to switch to any of the approaches during their whole journey based on the on-road conditions.

**Chapter 4** describes *Q-Safe*, a QoS-aware pricing model in which the low-price and high-price parameters suggest to end users. Prices are recommended for parameters that are impacted by two categories of parameters: dynamic and static.

**Chapter 5** addresses the problem of task overloading at the edge/cloud introducing edge servers near the edge nodes. The overall load is distributed among edge nodes, edge servers, and the cloud. In a growing number of on-road vehicles, a huge amount of time-sensitive data is uploaded to the edge node. Processing the data with limited storage and computing capability at the edge nodes is difficult. Introducing distributed edge servers near the edge layer tasks of edge and cloud layer are distributed among edge servers. It minimizes response time delay.

**Chapter 6** discusses a pricing scheme *Safe-Price*, in which optimized price is charged from the end users by minimizing processing cost and optimizing decision parameters. Also, SSPs make a profit after analyzing cash inflow and outflow.

**Chapter 7** addresses the problem of optimized safety plan generation and delivery of the safety plan to neighboring vehicles in minimum delay.

**Chapter 8** finally concludes the thesis by summarizing the previous chapters and envisioning the open scopes of future work in these domains.

# 2

# Related Work

This chapter presents an in-depth exploration of the existing literature in the field of provisioning safety-related services to on-road vehicles in the transportation sector. First, we discuss existing works regarding the impact of WSN and AI-based classification of risk factors for safety measures related to on-road vehicles in the transportation sector. Additionally, we discuss research works that address the challenges that arise in cloud-based, WSN-applied transportation. Next, we focus on the existing works on real-time safety-related issues in IoT-based transportation. We address the challenges of resource and pricing management in Safe-aaS and examine how existing research methodologies have attempted to resolve these issues to enhance safety for on-road vehicles in the road transportation industry. Our literature survey has three directions - identifying safety measures for on-road vehicles, existing resource management to provide safety-related information to on-road vehicles, and pricing mechanisms for provisioning on-road vehicular safety in a dynamic environment.

## 2.1 Safety Measures for On-Road Vehicles

Safety measures in on-road vehicles are crucial for preventing accidents and reducing the severity of injuries. Researchers investigated various risks associated with on-road driving and subsequently developed safety measures to address these concerns. This discussion highlights the work of earlier researchers who introduced safety measures after identifying different risks.

### 2.1.1 State-of-the Art: Risk Analysis of On-Road Vehicles

One of the primary reasons for sudden injury or death to human beings and financial losses is motorized vehicle accidents. Surveys indicated that the number of accidents is increasing daily. Most vehicle accidents result from poor driving practices. Sensor technology with provisioning alert messages to the drivers improves the driving experience in real-time environments. We categorize on-road risk factors into six categories- road condition, on-road traffic, driver's distraction, weather condition, vehicle malfunctioning, and network-based inefficiency as identified in Fig. 2.1.

***Risk Category 1: Road Conditions*** Road conditions significantly impact the safety, efficiency, and reliability of transportation systems. The risks associated with road conditions encompass a variety of factors, including surface quality, weather impacts, infrastructure design, and maintenance. Studies have shown that damaged road surfaces, such as potholes, cracks, and uneven pavement, increase accident risks due to loss of vehicle control and damage to vehicles. Researchers [27], [28] concentrated on the detection of potholes and speed breakers to identify risks for real-time support of vehicles. Ikeda *et al.* [29] described the detection of road surfaces after natural disasters. Three-axis accelerometers and GPS of the smartphone have been used to sense the road surface after natural disasters and rescue cars and people. The detection of pedestrians and the road surface has been achieved through sensing techniques. Satoh *et al.* [30] estimated road condition as the Angle of Inclination (AoI), evenness, and stability in places where pedestrians wear shoe-mounted sensors at the time of walking. A pair of sensors was attached to pedestrians' shoes to help them navigate various road surfaces. Researchers classified collected data into two categories of road surfaces - flat and other. On-board sensors might sense the road and traffic conditions to classify obstacles into two categories- static and moving objects to predict road situations. Chang *et al.* [31] introduced the framework of an interactive intelligent driving assistance system that can detect on-road obstacles. It helps to avoid road accidents. Another reason for road accidents is rough road surfaces. Excessive friction in rough surfaces causes sudden accidents. Du *et al.* [32] analyzed the sensed data by utilizing a deep neural network with the knowledge base to estimate the right friction on the road to avoid sudden clashes for autonomous vehicles. Road manholes are common in road transportation for various purposes such as passing rainwater, drainage, and power cables. If this cover is removed or broken somehow then it is difficult to maintain smooth driving at night specifically. Manhole cover detection is one of the major challenges in road surfaces. Yu *et al.* [6] proposed an automated algorithm on Mobile Laser Scanning (MLS) data and applied a supervised deep learning model to discover the manhole covers. Greg *et al.* [33] design a Bayesian belief network-based Road Safety Assessment System. They assess the safety performance of road designs by integrating technological solutions before development.

***Risk Category 2: Traffic Condition*** Safe speed is always safe for transportation and to avoid collisions or accidents. In prior research, IoT-based intelligent transportation systems used cost-effective cellular network-based speed estimation techniques. Chaturvedi *et al.* [34] proposed a cellular network-based multi-modal system to estimate edge level speed. Sharp lane departure is an issue for collision on-road transportation. A GPS-enabled vehicle with a smartphone is enough to estimate traffic parameters in the research works [35]. Several approaches have been proposed by researchers to identify unexpected lane departure events. When this situation occurs, drivers will receive a warning message to reduce the speed for the avoidance of accidents [36], [37].

***Risk Category 3: Driver's Behavior*** Two primary causes of road accidents and the associated financial losses are driver lack of attention and drowsiness. To improve safety and efficiency in road transportation, ITS is the technology that allows to control, monitor, and manage different elements of roads. Driver's driving measures are the major issue in the occurrence of on-road hazardous conditions. Although various ADASs [38] have highlighted and guaranteed safe driving on the road with certain limitations. Further, in the research work by Fazeen *et al.* [39], they described that smartphones equipped with limited sensors can improve driver safety. But driver's driving characterization [40] is measured in different aspects such as facial features, head movement, and eye movement. A fuzzy rule-based system classifies drivers' profiles based on behavioral characteristics. In survey works, [41], [42], researchers identified driver's drowsiness and distraction as the most important behavioral characteristics behind sudden accidents. They focused on smartphones and wearable devices for safe driving in Vehicular Adhoc Network (VANET)'s primary purpose of driving safely. ECG-based drowsiness detection scheme [43], classifies drivers' drowsiness by proposing a Drowsiness Detection Classifier (DDC) which helps to detect the stage of drowsiness and sends an alert message in that way.

***Risk Category 4: Weather Condition*** The impact of weather conditions on highway traffic is classified into three dimensions such as traffic safety, traffic flow, and traffic flow relationship. Agarwal *et al.* [44] described the impact of heavy rain and snowfall on urban roadways. They determine how variations in precipitation intensity affect the speed, headway, and capacity of roads. It categorized rain and snow occurrences according to their intensity levels. They define operating speeds and capacities to the different intensities of rain and snowfall. Unusual weather patterns also increase the risk of accidents on the roads. The main goal of traffic operations is to comprehend how road weather conditions affect the frequency and severity of crashes. The major risk factors are lack of clear visibility and high wind speed. Collision risk increases on roads in rural areas. Pavlou *et al.* [45] studied the impact factors of the on-road risks on rural roads due to adverse weather conditions. Weather condition-based driving simulator [46] detected eleven distinct weather conditions, encompassing clear skies, four categories of fog, four intensities of rainfall, and two classifications of snowfall.

Further, we discuss the technological solutions in the handling of safety services provided

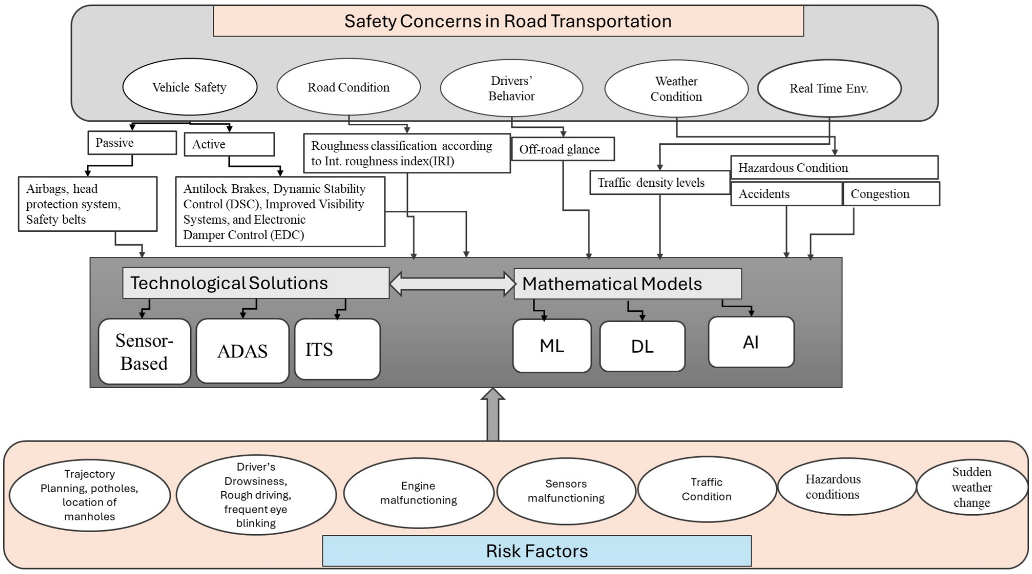


FIGURE 2.1: Safety and Risk factors

to on-road vehicles in automobile or transportation industries. These technologies are broadly classified as sensor-based and ITS-based [47]. In the above Fig. 2.1, it is shown that sensor-based technologies with various mathematical models improved IoT-based solutions to increase on-road safety. We concentrate on ITS-based solution approaches in IoV network. In IoV, wireless communications are established between vehicles (V2V), between vehicle and Infrastructure (V2I), and vehicle, and everything to deliver information about various risk factors related to the road, real-time environment, and vehicle malfunctions. It creates Social IoV (SIoV).

### 2.1.2 State-of-the Art: Technological Solution Approach

In SIoV, various ITS technologies such as Global Positioning System (GPS), Dedicated Short Range communications (DSRC), wireless communications, radio waves, infrared, and mobile technologies are used for communication, navigation, and connection. DSRC is a one-way or two-way communication channel with a 5.8GHz wireless spectrum. It is mainly used for V2V and V2I communications. In the SIoVs, vehicle nodes share and exchange data with various entities, forming a type of transient, socially conscious network equivalent to conventional social networks [48]. vehicles connect, and access the internet with 5G network [49]. Shenghai *et al.* [50] presented an integrated model in which cars had distance sensors in addition to DSRC, and GPS. The vehicle was notified to maintain a safe distance using distance sensors, that determine the distance between vehicles. Sensors in vehicles and roadside locations continuously monitor the environment, generating a vast amount of data in what is known as Data Driven Intelligent Transportation System (D2ITS) [50]. These systems allow users to access and utilize data resources related to transportation, providing more reliable and convenient services that enhance the overall performance of transportation systems. It is intended for cloud computing to enhance transport services and products in the automated sector. Since numerous automobiles are equipped with internet-accessible devices, Olariu et al. [51] suggested combining cloud computing, onboard devices in vehicles, a lot of sensors, and existing transportation networks to establish Vehicular clouds. AI and ML techniques are playing pivotal roles in enhancing the safety of on-road vehicles, especially in dynamic environments. These technologies enable real-time data processing and decision-making, which are crucial for ensuring the safety of autonomous and connected vehicles. This survey explores the applications of AI, and ML in this context, and focuses on recent advancements, methodologies, and case studies.

### AI and ML Techniques for Vehicle Safety

Existing research works reveal that machine learning models are widely used models to detect and reduce road accidents in order to improve on-road safety. In the state-of-the-art discussion, researchers [52], [53], [54] used various ML models to monitor traffic behavior and identify vehicles deviating from typical traffic patterns as potential indicators of road accidents. Saravanan *et al.* [55] developed a system with a proposed ML model to analyze potential safety concerns and deliver prior information to avoid accidents. Despite improvements in road and vehicle safety, traffic accidents continue to rise, highlighting the need for effective solutions. Researchers [56] introduced an intelligent traffic accident detection system leveraging VANETs, where vehicles exchange real-time data such as speed and coordinates to enhance road safety. Simulated data from VANETs is analyzed using supervised machine learning algorithms, including ANN, SVM, and Random Forest (RF). Reinforcement learning (Reinforcement Learning (RL)) algorithms, such as those discussed by Lillicrap *et al.* [57], are used for making real-time driving decisions in dynamic environments. Predictive models, such as those employing Recurrent Neural Network (RNN), and Long Short Term Memory (LSTM) networks, are used to forecast potential collisions and other hazardous events. Studies by Mozaffari *et al.* [58] highlight the use of predictive analytics in vehicular safety systems. AI and machine learning are increasingly used for predictive analytics, improving the accuracy and reliability of safety systems [59], [60].

### Predictive Modeling

- ***Accident Prediction Models*** Earlier researches reveal how ML algorithms, such as Support Vector Machine (SVM) and Neural Networks, can predict potential accidents by analyzing historical traffic data and real-time vehicle sensor inputs. There are several reasons identified by researchers such as driving behavior, harsh driving nature, a sudden street slopes. Researchers in [61], [62], analyzed the dataset which contains several attributes such as vehicle type, age sex, time of the day, and weather. They present a comparative study among various ML algorithms and drivers are alerted about the traffic accident risks.

- ***Driver Behavior Analysis***: One of the leading causes of car accidents and incidents is driver distraction. Tango *et al.* [63] analyzed visual distraction data of driver's driving and studied with various ML models. They estimate SVM as the best-fitted model for the dataset. Kaplan *et al.* [41] studied driving behavior and categorized it into two categories-driver's drowsiness and distraction. They consider both visual and non-visual features for classifications using traditional as well as modern ML models.
- ***Environmental Hazard Detection***: The Advanced Driving Assistance System provides safety-related information such as driving, driver, road conditions, and emergence of hazards. Kim *et al.* [64] proposed an active driving assistance system that provides collected information about the environment to the vehicles or the driver's smartphone. Various neural network models such as Backpropagation Neural Network (BP) network, and Neuro-fuzzy [65] were used to predict accidents based on real-time traffic conditions.

**Real-Time Data Processing in IoT-based Solutions**

- ***Sensor Based:*** Combining data from multiple sensors such as Light Detection and Ranging (LiDAR), radar, cameras enhances the accuracy of environment perception. Research by Chen *et al.* [46] explored sensor fusion techniques for autonomous vehicles, demonstrating improved object detection and tracking capabilities.
- ***Cloud-Based:*** One of the major issues in ITS, is the improvement of traffic assessments, and lowering the number of accidents. Sensors installed in vehicles or roadside, generate huge volumes of real-time data. Edge devices such as onboard sensors in vehicles, and RSUs are unable to process huge time-sensitive data at any instant. Cloud computing offers extensive computational resources for processing large volumes of data generated by vehicles. Studies by Zhang *et. al.* [27] examined the integration of cloud computing with vehicular networks for enhanced data analytics and decision-making. The emergence of the IoT and cloud computing provided an excellent opportunity to further address the growing transportation-related challenges, like congestion, heavy traffic, and vehicle safety. Recently, several approaches that leverage cloud computing to construct ITSs have been developed by researchers [66]. Cloud-based traffic management system [67] controled and optimizes real-time traffic conditions. In recent advancements, vehicle data is uploaded to the cloud platform, He *et al.* [68] proposed to use cloud computing and the IoT as an enabling infrastructure. Cloud-based vehicular safety leverages cloud computing to

enhance the safety features of vehicles through improved data processing [69], [70], communication, and scalability [71]. This includes integrating Vehicle-to-Everything (V2X) communication, autonomous driving technologies, and ADAS. The cooperative awareness between vehicles [72] reduces accidents in V2V communication significantly. Other communications such as V2I, and V2P with integrated cloud infrastructure [73], [74], [75] played significant roles in provisioning safety.

- ***Safety-as-a-Service:Safe-aaS*** Safe-aaS [19] platform is emerging as a unique cloud platform in the transportation ecosystem, requiring efficient pricing and resource management strategies to ensure timely and reliable service delivery. This survey explores existing literature on pricing models, resource management techniques, and their applications to real-time vehicular safety systems.

## 2.2 Issues in Resource Management for The Safe-aaS Platform

Safe-aaS is an IoT-enabled framework that enhances transportation safety by leveraging distributed sensors, data processing, and real-time decision-making. It is designed to provide customized safety-related information to on-road vehicles. It is a unique and common platform where several business entities such as sensor owners, vehicle owners, and drivers/end users, SSPs are involved. Safe-aaS is based on a service-oriented architecture with a pay-per-use basis. In Safe-aaS, decision virtualization is introduced to provide safety-related decisions to multiple end-users simultaneously. End users request the service by registering themselves in the web portal or application. Effective resource management in Safe-aaS ensures service quality, cost efficiency, and scalability. Below are the challenging aspects of resource management in Safe-aaS:

- ***Sensor Deployment and Utilization:*** Sensors are deployed in vehicles or along the roadside in the Safe-aaS platform. Roadside sensors are situated in various geographical locations with differing features. The effectiveness of Safe-aaS relies on optimizing sensor placement and ensuring redundancy. Sensors and devices should be adjusted based on data collection frequency, energy consumption, and real-time traffic or environmental conditions. There may be instances where redundant sensor data is being stored.
- ***Latency and Real-Time Processing:*** Safety services often require real-time data processing to ensure timely responses to emergencies or incidents. However, using cloud-based architectures might be challenging in this regard. Delays in processing safety, data may lead to catastrophic outcomes, such as accidents. If real-time processing is performed closer to the source such as vehicles, or road infrastructure, latency may be reduced. To

address this, it is beneficial to integrate edge computing, which processes time-sensitive data locally while utilizing cloud resources for more analytics and storage.

- ***Adaptability to Dynamic Environments:*** Safety requirements can change rapidly due to evolving risks, or new operational environments. Static systems are often less effective at responding to emerging threats or adapting to different safety requirements. Utilizing AI and machine learning to monitor trends can help to adjust safety provisions. Additionally, adopting flexible, modular architectures allows for better adaptation to changing environments.

- ***Scalability:*** Safe-aaS must be designed to scale to accommodate growth in users, devices, and data volume as clients' safety needs expand. If it fails to scale effectively, the service performance may degrade, and safety measures could be compromised. It's essential to design the cloud architecture with scalability in mind, utilizing load balancing and distributed computing to manage increased demand efficiently.

- ***Incident Response and Recovery:*** Safe-aaS service providers should establish comprehensive incident response and recovery plans to effectively manage emergencies or system failures. Inadequate incident response can exacerbate safety issues and prolong recovery time, potentially resulting in harm and damage to reputation. To manage incidents effectively, it is essential to implement automated alerting and response systems, conduct regular drills, and develop thorough recovery protocols.

- ***Pricing and Resource Allocation*** The provision of high-quality Safe-aaS is costly, as it requires investments in technology, training, compliance, and maintenance. High operational costs may affect the making of profit for the SSP. Optimization of resource allocation may reduce operational expenses with the cost-effective cloud solution. Dabbagh *et al.* [76] proposed an online pricing scheme for resource allocation in the cloud minimizing energy consumption, and maximizing the profit of the service provider. In a dynamic environment, region-wise road conditions vary. Price charged from users vary region-wise as safety-related parameters may differ. The RegPrice [77] framework demonstrates cost-efficiency and fairness in IoT-based safety services with varied costs due to geographic and infrastructural differences. Dynamic or pay-as-you-go models [78] adjust the price according to system demand or risk level. Studies in research works [11], [79], use game theory to model pricing decisions in a competitive environment where multiple Safe-aaS providers operate. These models help to optimize pricing based on factors such as bargaining between SSPs and users, demand for service, and service quality.

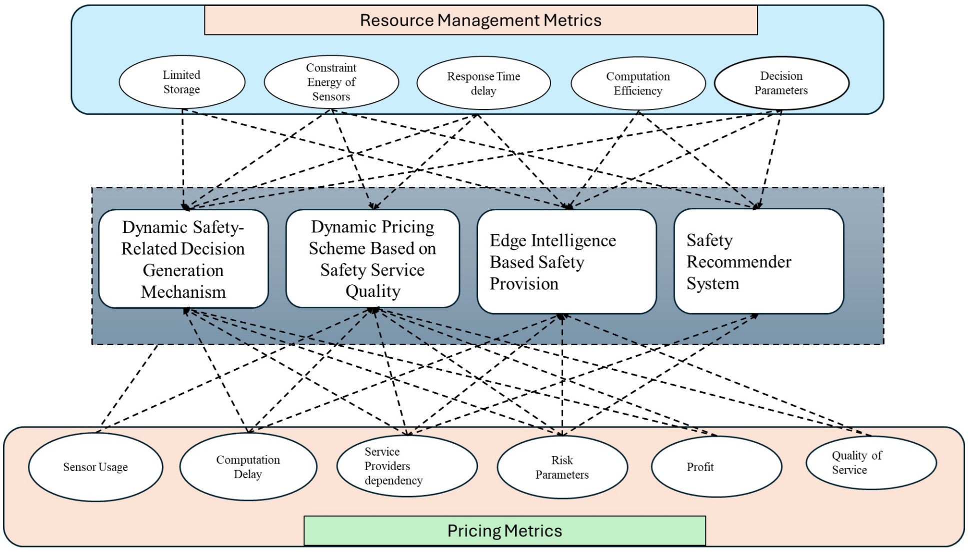


FIGURE 2.2: Proposed Solution Approach

## 2.3 Summary

There are two main perspectives to consider: real-time safety for on-road vehicles and the business aspects of the transportation industry. For ensuring safety, resource management is a critical issue within the Safe-aaS platform. Edge devices and servers all have limited resources, and the increasing number of vehicles on the road at any given time creates hazardous conditions such as congestion and road accidents. This makes it challenging to manage and deliver timely and accurate safety-related decisions. Our extensive solution approaches address the management of resources such as storage space, computational capability, and response time delays.

In the Safe-aaS platform, end users pay for the services they utilize based on their specific needs. The primary requirement for these end users is the provisioning of high-quality service. On the other hand, SSPs make a profit after deducting the payments owed to sensor vendors or other SSPs from the revenue generated from the end users. QoS is dependent on effective resource management within Safe-aaS. We propose solution approaches as in Fig. 2.2 for effective resource management and pricing management.

# 3

# Dynamic Adaptive Decision Generation

This chapter presents "Dec-Safe," a dynamic decision-generation framework for effectively managing resources like cloud storage, sensor usage, response time delays, and service utility. This is an adaptive decision-generation mechanism for the Safe-aaS platform. End users receive safety services as safety-related decisions after the selection of decision parameters by logging in to the portal. We categorize decision parameters as static and dynamic for the two approaches - static decisions and dynamic decisions respectively. Automation has been adopted throughout numerous industrial sectors in recent years through the combination of traditional and state-of-the-art technology. Researchers proposed various mechanisms [80–82], and real-time assistance system [19, 83–89] in the field of road transportation. To reduce the number of traffic accidents, the road transportation industry depends on the safety of its drivers and cars. Various vehicular technologies, including ITS and ADAS, are implemented in vehicular networks to enhance low-latency traffic flow. Moreover, AI-based techniques are employed to improve on-road safety for vehicles [11]. One of the uniquely designed platforms that serves many users with dynamic safety-associated decisions is Safe-aaS. Considering road transportation as the Safe-aaS implementation scenario, we create a decision-making process to ensure safety.

The users enter their starting and destination locations while registering for the on-road safety services in Safe-aaS, choose some decision variables, and pay through the Web portal. We introduce static and dynamic approaches for availing the safety services. Depending upon the selected approach and chosen decision variables, a decision is transmitted to the users. In the static approach, the static parameters, whose values do not vary regularly with time, are suggested to the users. The users have the flexibility to select additional parameters as per their requirements in a real-time environment. The parameters are associated with certain decisions. On the other hand, in the dynamic approach, users can alter the selected parameters at any time instant until reach the destination.

In our proposed decision-generation mechanism, "Dec-Safe", we evaluate and analyze the cost incurred by the users for availing the static and dynamic decision-generation services. Several strategies have been proposed in the research works on road safety to reduce accidents and enhance driver safety, including on-board training [86], road manhole cover detection [88], and traffic management in smart cities [87]. Safe-aaS is one of the developed schemes to provide safety-related decisions to the end users in a dynamic environment. Sometimes users select certain decision parameters with similar characteristics. At any time instant, a huge volume of registered users requests for similar decision parameters which leads to huge energy loss of the sensors as the sensing, transmission, and processing of these similar decision variables consume unnecessary energy and time. Safety decisions are time-sensitive and any delay in the delivery of the decisions may cause unwanted incidents or accidents. The sensor nodes are also energy-constrained in nature, therefore, it is necessary to conserve the energy for future applications. Decision generation incurring minimum delay is desirable to avoid road accidents. Owing to the above challenges, we develop a dynamic decision-generation mechanism in the Safe-aaS platform. As Safe-aaS provides safety-related decisions on a pay-per-use basis. In this decision-generation mechanism, users have the flexibility to choose any of the decision-generation mechanisms as per their requirements during the entire journey. We propose static and dynamic approaches to avail safety services by the end users. By mapping the decision variables with the generated decisions in a static approach, we minimize the quantity of cloud storage utilized and computing complexity. In the dynamic approach, we present the concept of decision clustering and apply maximum likelihood estimation on correlated decision variables within the users' selected parameters. Finally, generated decisions are delivered to the end users. The same decisions are not generated repeated times for multiple users in the decision clustering method. As a result, the suggested parameters give consumers the flexibility to choose the safety services that best suit their needs in customized costs. In this chapter, we try to address the following issues that arise in the safety-related decision-generation mechanism.

- ***Provision of flexibility in safety services***: Our adaptive decision generation mechanism, "Dec-Safe", provides safety-related decisions to the users. It consists of two approaches- static and dynamic. End users possess the flexibility to choose any of the approaches. In the static approach, users select decision parameters whose status and values do not change frequently with time. On the other hand, in the dynamic approach, parameters change their status with time and it depends on the dynamic on-road environment. During the entire journey, users may change the mode of decision-generation mechanisms. As Safe-aaS is a pay-per-use service then customized price is calculated against service usage by end users with adaptability.

- ***Minimization of memory space needed at the cloud server to generate decision applying our static decision generation mechanism***: We propose a modified Boolean multiplication operator to generate decisions for the static approach. First, we group the users who select similar decision variables. Thereafter, we define a mapping function to ensure the delivery of these decisions simultaneously, to multiple users. We design a utility function for users and formulate a min-max optimization problem, such that the similarity among the decision variables is maximized when the optimum volume of memory space is utilized.

- ***Minimization of the usage of sensor nodes, computing time and response time delay with our dynamic approach***: In the case of the dynamic approach, we apply a clustering approach, which is performed by finding the maximum likelihood probability estimator for similar selected decision variables. The generated decision is provided to the users, based on clustering. The limited number of sensors is used in the clustering process. Sensor usage is almost constant in our approach. We calculate the utility per user and formulate an optimization problem to minimize the utility for an optimal value of response time delay and effective residual energy.

- ***Reduction of energy consumption by the sensor nodes for our scheme***: Exhaustive analysis of the results of "Dec-Safe" signifies that the overall cost incurred in the case of the static approach is less compared to the dynamic approach. Additionally, regression analysis of the average storage, effective cost (static and dynamic approach), utility, and effective energy illustrates the future variation in the trend of these parameters.

- ***Minimization of delay in the decision delivery:*** Response time is one of the important factors in providing safety-related decisions to the users. We provide extensive simulation results of response time delay for the users. In our dynamic approach average response time is increased in increasing trend of user range but per user delay maintains a constant/ decreasing pattern in the certain number of iterations. The reason behind the pattern is time to deliver the decision decreases as clustering progresses.

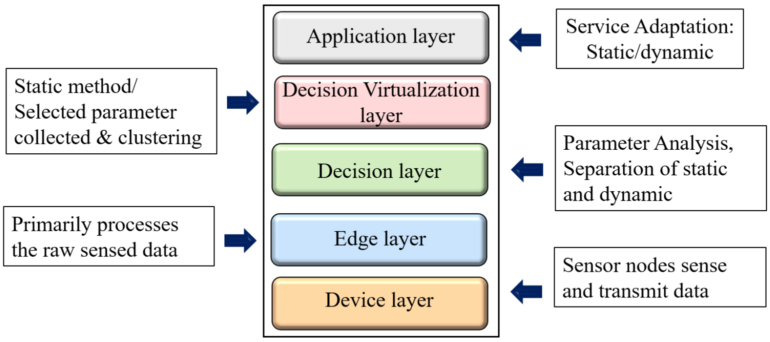


FIGURE 3.1: Dec-Safe: The System Architecture

## 3.1 Decision Making In Safe-aaS Infrastructure

Safe-aaS is the widely used infrastructure in the ITS application scenario. It consists of five layers- device layer, edge layer, decision layer, decision virtualization layer, and application layer. Heterogeneous types of sensor nodes are present in the device layer. Sensors are deployed in vehicles or placed on the roadside. These sensors sense the environment and transmit to the edge layer/cloud layer based on the time sensitivity. The primarily processed data are transferred to the decision layer for the decision-generation process. Both the decision layer and decision virtualization layer are responsible for the decision generation. The generated decisions at the decision virtualization layer are logically mapped to the decision variables that the end users have requested. Users create an account on the Safe-aaS platform through the web portal to access safety services on a pay-per-use basis. Certain chosen variables in the real-world scenario might not change over time in a dynamic way. These parameters are regarded as static decision factors, whereas dynamic decision variables are those that vary with time. The layer-wise modified system architecture of "Dec-Safe" is described in Fig. 3.1.

The generated decisions are logically mapped with decision variables. In our scheme, decision variables are categorized into two categories- static and dynamic. The detailed decision generation mechanism is shown in Fig. 3.2, End users have the flexibility to choose either type of variable. Based on the selection of variables, any of the decision generation modes is activated and price is charged from them. Users select static parameters at the beginning of their journey and these parameters are not alterable. The static decisions are generated when the end users select static parameters and when users select dynamic parameters during the journey, dynamic decisions are provided to the users. We apply a decision clustering approach to generate dynamic decisions based on the decision parameters selection.

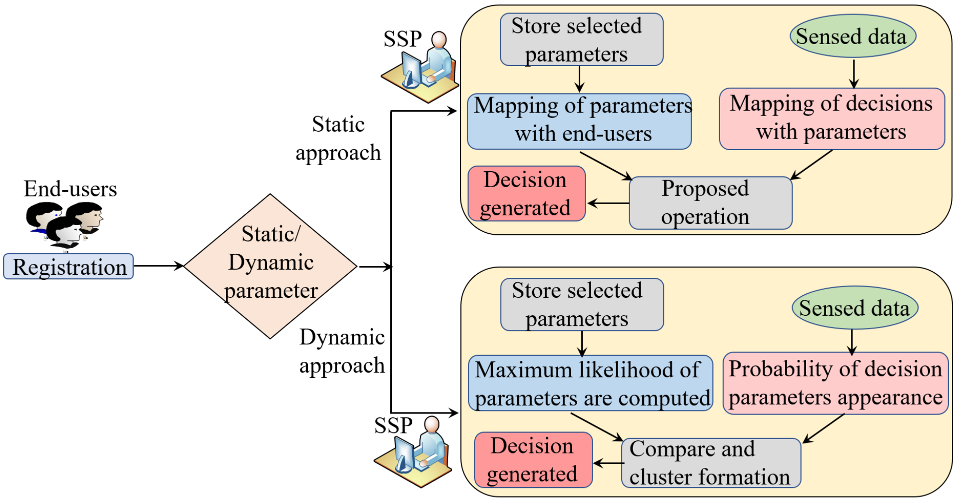


FIGURE 3.2: The Dec-Safe: Workflow Diagram

## 3.2 Dec-Safe: Mathematical Model

We consider $D^t = \{d_1, d_2, \cdots, d_N\}$ as a set of $N$ decision variables, such that $d_i \in D^t$ and $1 \leq i \leq N$, at any time instant $t$. These decision variables are either static or dynamic in nature. Therefore, $D^t = D^{s,t} \cup D^{dy,t}$, where $D^{s,t}$ and $D^{dy,t}$ represent the subsets of static and dynamic decision variables, respectively. We denote any $i^{th}$ generated decision as $\mathbb{D}_i$. Further, the set of registered users, $\mathcal{E} = \{E_1, E_2, \cdots, E_n\}$, requests for the safety-related decisions. Any of the $j^{th}$ user, $E_j$, may select $k$ decision variables from the set $D^t$. Additionally, $\mathcal{E} = \mathcal{E}_s \cup \mathcal{E}_{dy}$, where $\mathcal{E}_s$ and $\mathcal{E}_{dy}$ represent the set of users requesting for static and dynamic decision variables, respectively. The users select the type of service - static and dynamic - during registration, as per their requirement, on payment basis.

### 3.2.1 Static Approach

The users who select static parameters during their registration, are provided with static services by the SSP. We represent the set of static decision variables as $D^{s,t}$=$\{d^s_1, d^s_2, \cdots, d^s_k\}$, where any parameter ${d^s}_i \in D^{s,t}$, such that $1 \leq i \leq k$ and $D^{s,t} \in D^t$. Any user $E_i \in \mathcal{E}_s$, such that $\mathcal{E}_s \subseteq \mathcal{E}$. As the value of decision variables does not fluctuate frequently with time, therefore, the decision generated with these static-type parameters are temporarily stored in the decision layer. To minimize the computation time and storage space required, we map these generated decisions with the static parameters requested by the users. Further, we represent this mapping between the stored decisions and the requested decision variables in the form of a matrix, $\mathcal{M}_{dp}$. We mapped the decision variables demanded by each of the users with the total registered users at a particular time instant to find the overlapping decision variables. Mathematically,

$$\mathcal{M}_{es}[i][j] = \begin{cases} 1, & E_i \text{ selects } d_j^s \\ 0, & \text{otherwise} \end{cases} \tag{3.1}$$

Thereafter, the mapping of the requested decision variables with the decisions generated is represented as,

$$\mathcal{M}_{dp}[j][k] = \begin{cases} 1, & \text{if } d_j^s \text{ is required for decision } \mathbb{D}_k \\ 0, & \text{otherwise} \end{cases} \tag{3.2}$$

where $\mathbb{D}_i$ represents the generated decision. We consider that the users need to select a minimum $P$ number of decision variables, where $P$ is a non-zero integer value set by the SSP. To find the final mapping, $\mathcal{M}_D$ of the users with the decision variables requested by them, we apply a Boolean function. We introduce a modified Boolean multiplication operator combining two different logical operators, which is mathematically expressed as,

$$\mathcal{M}_D[x][y] = \mathcal{M}_D[x][y] \oplus (\mathcal{M}_{es}[x][z] \wedge \mathcal{M}_{dp}[z][y]) \tag{3.3}$$

where $1 \leq x \leq m$, $1 \leq y \leq l$ and $1 \leq z \leq k$. As the function, $\mathcal{M}_D$ generates decision(s) based on the users selected decision variables, the response time and memory usage is minimized per response by grouping those decision variables with the similar decisions. We represent $\mathbb{E}_D$ as the set of end users, whose demanded decision variables match any specific decision $d_i^s$, such that $\mathbb{E}_{D_i} \subseteq \mathbb{E}_D$.

***Definition* 1.** *We define effective memory space required to process the decision variables chosen by any user as a ratio of the residual memory of the server at the time instant, t to the total available memory. Mathematically,*

$$M_j = \frac{M_{R_j}}{M_{tot}} \forall j \in E_s \tag{3.4}$$

*where $M_{R_j}$ is the residual memory of the server after serving the $j^{th}$ user and $M_{tot}$ is the total available memory.*

***Definition* 2.** *We map the decision variables used to generate a decision for the $j^{th}$ user with the set of available decision variables. We termed this function as effective parameter matching, which is represented as,*

$$F(d_{s_j}, d_j) = \frac{\sum_{i=1}^{k} d_{s_{ij}}}{\sum_{r=1}^{D} d_{rj}} \forall j \in E_s \tag{3.5}$$

*where $d_{s_{ij}}$ and $d_{rj}$ is the set of decision variables selected by the $j^{th}$ user and the ones available, respectively.*

**Algorithm 1** Static Approach

```
INPUT: ⟨source, destination⟩ = $\langle S_i, d_i \rangle$, $E_S$, $d_{S,t}$
OUTPUT: Decision $D_i$ is delivered to multiple users
PROCEDURE:
for $m$ End users, $k$ decision variables do
    Map each $d_{S,t}$ with each $E_S$ in $\mathcal{M}_{ES}$;
end for
for $l$: Decisions, $N$: decision variables do
    Map each $d_{S,t}$ with each decision $D$ in $\mathcal{M}_{dp}$;
end for
for $m$: End users and $l$: Decisions do
    $\mathcal{M}_D$ mapping: operation on $M_{es}$ and $M_{dp}$: decision generated;
    while time = $\tau$ do                                  ▷ $\tau$: short time duration
        SSP suggests $j$ parameters $\longrightarrow$ $i^{th}$ user - $\langle S_i, d_i \rangle$;
        if $i^{th}$ user agrees then
            Decision generated;
            Price and Utility is estimated;
        else
            Select new set of parameters;
        end if
    end while
end for
```

The primary objective of this work is to reduce the volume of memory utilized for the decision generation. To achieve this, we maximize the parameter matching between the selected and available set of decision variables. We define a resource utilization function $\mathcal{U}_j$ for the $j^{th}$ user,

$$\mathcal{U}_j = \alpha F(d_{s_j}, d_j) \times \beta M_j \tag{3.6}$$

where $\alpha$ and $\beta$ are pre-defined constants used for application-specific requirements. We consider that the value of these constants are user-defined and $0 < (\alpha, \beta) \leq 1$. Further, we formulate the utility of the user as a Linear Programming Problem. Our primary aim is to minimize the optimal memory space required for decision generation and maximize the matching of decision variables. Mathematically,

$$\underset{M_j}{\text{argmin}} \max_{F} \quad \mathcal{U}_j \tag{3.7}$$

subject to, $p \leq d_s \leq k$ and $0 < F \leq 1$. $p$ represents the minimum number of decision variables to be selected by the $j^{th}$ user. First, we maximize the matching of parameters to minimize the response time delay for any decision. Thereafter, we minimize the utility of any user for an optimal value of effective memory space. This is because both the decision variables and generated decisions are stored in the database of the server, in case of static approach.

### 3.2.2 Dynamic Approach

In the dynamic approach, the decision is generated for the users considering the dynamic fluctuations in the value of decision variables. Further, based on the requirements of the users, the decisions provided to them are updated in real-time. Therefore, the decision generation process is quite complex due to the dynamic variations in the value of the decision variables. We aim to optimize the response time delay and residual energy to generate the decision. As the sensor nodes are energy-constrained in nature, therefore it is necessary to preserve their energy for future applications. Considering these aspects, we apply clustering mechanism to group the similar decision variables selected by the registered users at any time instant, $t$. We apply maximum likelihood function to find the probability of the selected decision variables. Suppose, $D_{dy} = d_1, d_2, \cdots, d_n$ is the set of sample decision variables at any time instant, $t$ selected by any user, such that $D_{dy} \subseteq D^{s,t}$. The category and number of parameters in the set $D_{dy}$ may vary with time and the requirement of the user till he/she reaches the destination. Suppose, $\mathcal{P}$ is the probability of the emergence of the decision variables requested by each user. We assume that the decision variables chosen by users are normally distributed with respect to the generated decision. The probability density function is represented as $f(d_1, d_2, \cdots, d_n|p)$ where $p$ is the total probability of parameters. We represent the individual probability of the $i^{th}$ decision parameter, $d_i$, as $\mathcal{F}(d_i|p)$,

$$d_i = \begin{cases} 1, & \text{if } d_i \text{ appears in parameter space} \\ 0, & \text{otherwise} \end{cases} \tag{3.8}$$

and the probability as, $\mathcal{F}(d_i|p) = p^{d_i}(1-p)^{(1-d_i)}$. Therefore, the probability of the appearance of any decision parameter in parameter space is, $\mathcal{F}(1|p) = p(1-p)^{(1-1)} = p$, and the probability of disappearance is represented as, $\mathcal{F}(0|p) = p^0(1-p)^{(1-0)} = (1-p)$. The total probability of $n$ selected decision variables is expressed as,

$$\begin{aligned} \mathcal{F}(d_1, d_2, \cdots, d_n|p) &= p^{d_1}(1-p)^{1-d_1} \cdots p^{d_n}(1-p)^{1-d_n} \\ &= \prod_{i=1}^{n} p^{di}(1-p)^{1-di} \end{aligned} \tag{3.9}$$

Further, in case of similar selected decision variables from the user end, the joint probability is,

$$\mathcal{P}(X_1 = d_1 \cdots X_n = d_k) = \prod_{i=1}^{n} p^{di}(1-p)^{1-di} = L \tag{3.10}$$

where $R \leq k \leq n$. $R$ represents the minimum number of decision variables to be selected by the user at any time instant $t$. $L$ denotes the likelihood of similar decision variables and is mathematically expressed as, $L = \prod_{i=1}^{n} p^{X_i}(1-p)^{(1-X_i)}$. Additionally, $L$ attains the maximum value when, $\frac{dL}{dp} = 0$, therefore, $\frac{dL}{dp} = 0 \geq \mathcal{P}$. Safety-related decisions generation is continuous process in dynamic environment. Considering this fact, we apply the logarithm function to estimate the maximum likelihood of the decision variables. Therefore,

$$\begin{aligned} l_k &= \log \mathbb{L} = \log \big( \prod_{i=1}^{n} p^{X_i}(1-p)^{(1-X_i)} \big) \\ &= n\bar{X}_i \log p + n\big(1 - \bar{X}_i\big) \log(1-p) \end{aligned} \tag{3.11}$$

As derived in Equation (3.7), we calculate the likelihood estimator, after equating the first-order derivative of Equation (3.11) to zero.

$$\frac{\partial l_k}{\partial p} = n\bar{X} \log p + n(1-\bar{X}) \log(1-p) = 0 \tag{3.12a}$$

$$\frac{n \times \bar{X}}{\mathcal{P}} - \frac{n \times (1-\bar{X})}{1-\mathcal{P}} = 0 \tag{3.12b}$$

Therefore, $\mathcal{P} = \bar{X}$. Suppose, the probability of any decision generation is $P$. We calculate the maximum likelihood estimator from the set of safety-related parameters demanded by the users as the mean value of the set. Based on the number of decisions generated, the formation of the number of clusters is determined by the SSP. Suppose, $N$ clusters are formed at a particular time, $t$, is denoted as a set $\mathbb{C}^t = \{C_1, C_2, \cdots, C_N\}$. Therefore, any cluster $C_i$ is represented as, $C_i = d_i \in D^{dy,t} | C_i(d_i) = 1$, where $d_i$ is the decision parameter selected by the $i^{th}$ user. $d_i$ is randomly selected to minimize the error between the maximum likelihood estimator and the probability of any generated decision. The objective function to choose the decision variables for $C_i$ is,

$$G(C_i) = \frac{1}{N} \Sigma_{i=1}^{N} (\mathcal{P} - \mathcal{P}_i)^2 \tag{3.13}$$

In the dynamic approach, the number of clusters is not defined at the beginning of the decision generation. The number of clusters formed till the mean squared error, $e$ is almost equal to 0. Each of the clusters is formed from similar selected decision variables. Initially, the clusters are formed, thereafter, we apply the concept of decision virtualization for providing safety services to multiple users.

***Definition* 3.** *We define the effective residual energy of the $j^{th}$ sensor node as the ratio of the leftover energy of that sensor node to its initial energy.*

**Algorithm 2** Dynamic Approach

```
INPUT: $\langle$source, destination$\rangle = \langle S_i, d_i \rangle$, $E_{d_i}$, $D^{dy,t}$
OUTPUT: $D_i$
PROCEDURE:
    for Source to Destination do
        for Each user $E_{d_i}$ do
            if $E_d$== newly registered then
                for Each sample set of selected parameters do
                    Find maximum likelihood,$\mathcal{P}$.
                    for Each decision $D_i$ do
                        Generate random probability, P.
                    end for
                    Calculate $e$ between $P$ and $\mathcal{P}$.
                end for
                if $e \simeq 0$ then
                    Form cluster, $C_i$.
                end if
            end if
        end for
        Decision is generated.
    end for
```

$$f(\mathcal{N}_S, R^{eff}) = \mathcal{N}_S \times \Sigma_{j=1}^{\mathcal{N}_S} R_j^{eff} \tag{3.14}$$

Further, $\mathcal{R}_j^{eff} = \frac{R_{t,j}^{lo}}{R_j^{ini}}$ where $\mathcal{R}_j^{eff}$ is the effective residual energy of the $j^{th}$ sensor node and $\mathcal{N}_S$ is the number of sensor nodes required for the generation of the decision. Further, the leftover energy and initial energy of the $j^{th}$ sensor node is represented as $R_{t,j}^{lo}$ and $R_j^{ini}$ respectively.

***Definition* 4.** *We define the response time delay required to generate the $i^{th}$ decision as, $\mathcal{T}_{rs,i} = (T_{c,i} + T_{p,i})$, where $T_{c,i}$ is the time required to collect the sensed data and $T_{p,i}$ is the time required to process the data.*

In Safe-aaS platform, decision is received by the users purely depending upon the chosen parameters. We assume that there exists a upper and lower limit for the selection of the parameters. Further, the number of sensor nodes used for decision generation is optimized in the dynamic approach. We define an utility function in terms of the effective energy of the sensor nodes and the response time delay. Mathematically,

$$\mathcal{U}_j = \frac{1}{\mathcal{T}_j}\sum_{i=1}^{N}\left(f(\mathcal{N}_S^i, \mathcal{R}_i^{eff})\mathcal{T}_{rs,i}^x C_i\right) \tag{3.15}$$

where $\mathcal{T}_j$ is the total time required by the $j^{th}$ user to reach the destination. $x$ represents the factor that signifies the importance of the parameter $\mathcal{T}_j$. We formulate an optimization function to minimize the response time delay and residual energy of the sensor nodes.

$$\underset{\mathcal{T}_{rs,i},\mathcal{R}_i^{eff}}{\text{Minimize}} \quad \mathcal{U}_j \tag{3.16}$$

subject to $\mathcal{N}_S \geq k$, $0 \leq \mathcal{R}_j^{eff} \leq 1$, and $\mathcal{T}_{rs,i} \leq \mathcal{T}_j$. The minimum number of decision variables to be selected by any user is represented as $k$. The Lagrangian form of given optimization function in Equation 3.16 is represented as:

$$\mathcal{L}_j = \mathcal{U}_j - \mu_1(\mathcal{N}_S - k) + \mu_2(1 - \mathcal{R}_i^{eff}) + \mu_3(\mathcal{T}_j - \mathcal{T}_{rs,i}^x) \tag{3.17}$$

where $\mu_1$, $\mu_2$, and $\mu_3$ are the Lagrangian constants. Further, we solve this Lagrangian form using Karush-Kuhn-Tucker (KKT) conditions such as *dual feasibility* and *slackness conditions.*

$$\frac{\partial \mathcal{L}_j}{\partial \mathcal{T}_{rs,i}} = \frac{1}{\mathcal{T}_j}\sum_{i=1}^{N} f(\mathcal{N}_S^i, \mathcal{R}_i^{eff})\mathcal{T}_{rs,i}^{(x-1)} C_i - \mu_2 = 0 \tag{3.18a}$$

$$\frac{\partial \mathcal{L}_j}{\partial \mathcal{R}_i^{eff}} = \frac{1}{\mathcal{T}_j}\sum_{m=1,m\neq i}^{N} f(\mathcal{N}_S^m, \mathcal{R}_m^{eff})\mathcal{T}_{rs,i}^{x} C_i - \mu_3 = 0 \tag{3.18b}$$

$$\mu_i X = 0, \text{ and } \mu_i \geq 0, \forall i = \{1, 2, 3\} \tag{3.18c}$$

Therefore, the optimal value of the response time delay and effective residual energy are as follows:

$$\mathcal{T}_{rs,i}^{*} = \left(\frac{\mu_2 \mathcal{T}_j}{\sum_{i=1}^{N} f(\mathcal{N}_S^i, \mathcal{R}_i^{eff}) C_i}\right)^{\frac{1}{(x-1)}} \tag{3.19a}$$

$$\mathcal{R}_i^{eff,*} = \left(\frac{\mu_3 \mathcal{T}_j}{\sum_{m=1,m\neq i}^{N} f(\mathcal{N}_S^m, \mathcal{R}_m^{eff}) C_i \mathcal{T}_{rs,i}}\right) \tag{3.19b}$$

### 3.2.3 Cost Analysis

In Safe-aaS, the users choose certain safety-related parameters and make payments through a web portal. Therefore, the users only pay for the services availed by them, as per their requirements. As static and mobile sensor nodes are present in the device layer, sensed data is transmitted to the cloud/edge nodes. We consider that two types of services are availed by the users – static and dynamic.

In the static approach, the price charged from the users is calculated in terms of the

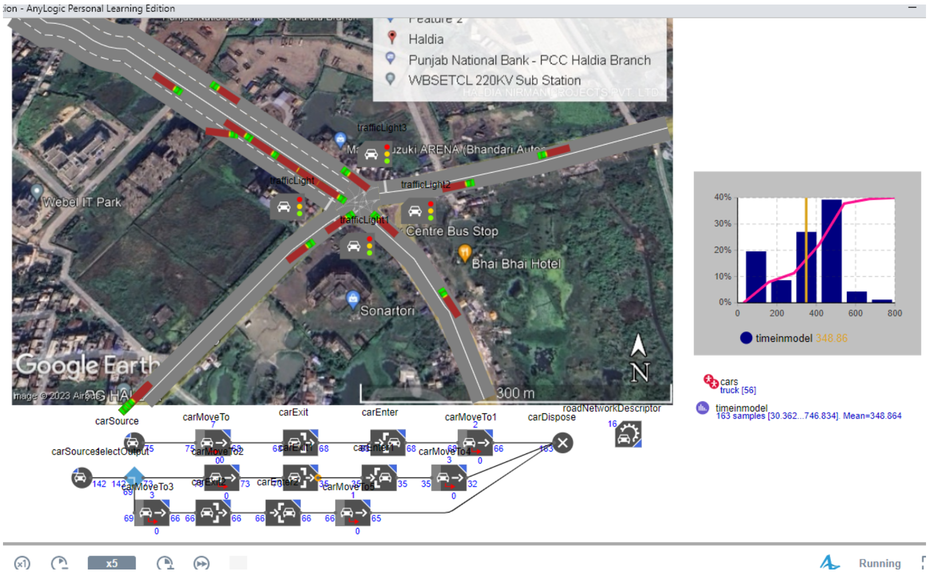
TABLE 3.1: Simulation Parameters of Dec-Safe

| **Parameters** | **Values** |
|---|---|
| System reserve memory | 512 units |
| $C_{min}$ | 4 units |
| $P_{min}$ | 1000 units |
| $P_{add}$ | 1000 units |
| $C_d$ | 1000 units |
| $D_{tot}$ | 10000 units |
| $D_{min}$ | 500 units |
| $N_s$ | 1 – 600 units |
| $C_{po}$ | 100–200 units |
| $R_{init}$ | 1 – 100 units |

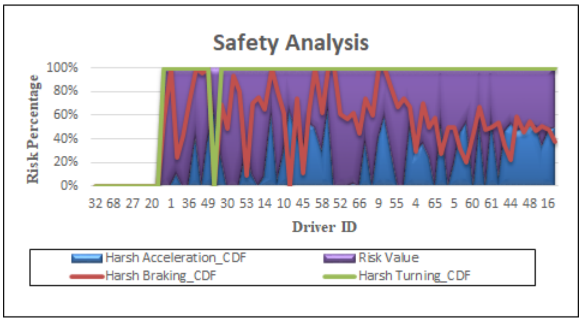
minimum number of parameters to be picked up by them, $p_{min}$, and the minimum per unit cost of these parameters, $C_{min}$. Further, if the user has chosen a static approach and requires some additional information, he/she may request the decision variables from the service provider. The supplementary decision variables to be provided are denoted as $p_{add}$ and the per unit cost of these decision variables is $C_d$. Suppose, $\mathbb{D}_a$ is the distance for which the Safety Service Provider (SSP) charges the minimum price per unit of the decision variables. In case any user requests safety services for a distance, $\mathbb{D}_t$, such that $\mathbb{D}_t > \mathbb{D}_a$, per unit cost of decision variables is $C_d$ ($C_d > C_{min}$). Motivated by the concept of penalty cost discussed in Safe-aaS platform, we consider that penalty cost, $C_{op}$, is given by the SSP, if the safety services are provided with a response time delay, $\mathcal{T}_{rs}$ of $\mathcal{T}_{rs} > \mathcal{T}_{allw}$. $\mathcal{T}_{allw}$ is the maximum allowable time duration beyond which the penalty cost is charged. Mathematically,

$$\begin{aligned} C_{st} = C_{min} \times p_{min} + C_{ad} \times p_{add} + C_d \times (\mathbb{D}_t - \mathbb{D}_a) \\ -C_{op}\frac{(\mathcal{T}_{rs} - \mathcal{T}_{allw})}{60} \end{aligned} \tag{3.20}$$

In the dynamic approach, we consider that $X^i$, $Y^i$, and $Z^i$ are the number of decision variables initially requested, added later, and reduced, by the $i^{th}$ user, while traveling from the source, $sr$ to the destination, $des$. We consider that the values of these decision variables may be updated with time by the user. Hence, we sum these fluctuations in the number of decision variables added or reduced by the user, and the price charged as $C_{dy}$.



(a) Mobility Region



(b) Analysis of Safety

FIGURE 3.3: Mobility Region of Vehicles and Safety Assessment

## 3.3 Performance Evaluation

### 3.3.1 Simulation Design

To evaluate and analyze the performance of our proposed scheme, "Dec-Safe", we use Python and Anylogic as the emulator tools to simulate our environment. We consider a simulation area of $10 \times 10km^2$, 10 types of decision variables, and 5 types of generated decisions. We perform the simulation of our proposed mechanism, Dec-Safe in two phases. In the first phase, we model the mobility of the vehicles applying Anylogic and in the second phase, we model the safety analysis using Python. Anylogic is a multimethod modeling tool that is applied in different sectors such as road traffic, healthcare, and supply chain management, to design business-oriented models. Additionally, Anylogic helps to visualize the scenario with the user interface, defines the traffic flow, and analyzes the real-time data generated from the model. Further, we consider two approaches - static and dynamic - using which the users avail safety

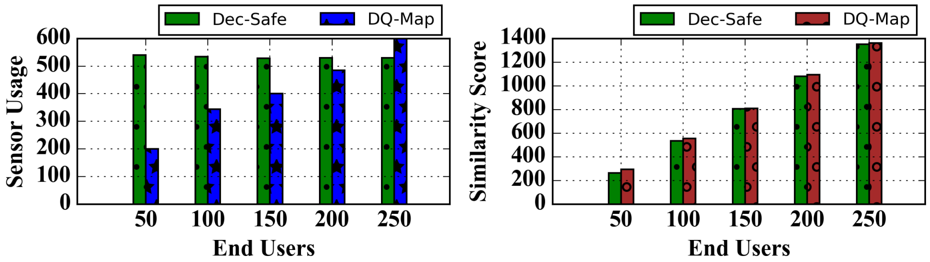


FIGURE 3.4: Benchmark Solution for Static Approach

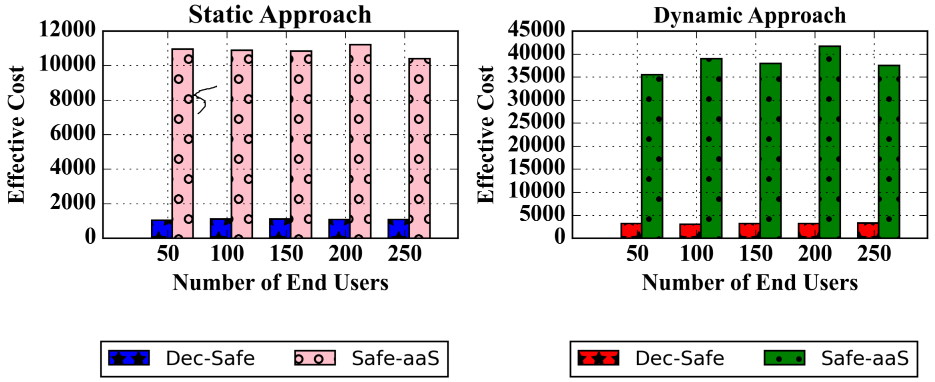


FIGURE 3.5: Variations in Effective Cost

services from the Safe-aaS platform. The static sensor nodes are deployed at a particular geographical location, while the mobile sensor nodes are placed in vehicles. Fig. 3.3 illustrates the mobility region of vehicles within the simulation area. We use the Anylogic platform to estimate the mobility of vehicles and their response time. We consider the traffic at a particular geographical location - Haldia, West Bengal, India. We utilize the open available driving dataset to analyze the safety of drivers, considering various risk factors such as the motion of the vehicle, distance from neighboring vehicles, and time left to reach their destination. We apply the dynamic clustering algorithm to estimate the risk above which the drivers are provided safety decisions. In the case of motion-based factors, we consider sudden break, acceleration, and sharp turns. Exhaustive regression analysis for every parameter is performed to predict the future trend of the results as we consider a random number of parameters and users. The details of the simulation parameters are given in Table 6.1.

$$C_{dy} = C_{min} \times p_{min} + \sum_{sr}^{des} \sum_{t=1}^{n} (X_t^i + Y_t^i - Z_t^i) \times C_p - C_{op} \tag{3.21}$$

### 3.3.2 Benchmark

We compare the proposed decision-generation mechanism with the existing traditional Safe-aaS platform. Roy *et al.* [19] proposed a unique platform, the first of its kind, for provisioning safety-associated decisions to users. They theoretically designed the platform and analyzed the transactions among the actors of Safe-aaS. However, they did not consider the memory space, residual energy, and utilization of sensor nodes for decision generation. We also compare the static approach of our proposed scheme with DQ-Map [90] in Fig. 3.4. The authors in DQ-Map considered the decision variables requested by users as decision queries and proposed a decision query mapping mechanism to minimize the usage of sensor nodes and decision generation time. They find the similarity score among the requested decision variables. We observe that as the similarity score increases, the usage of sensor nodes is also increased, in the case of DQ-Map. On the other hand, in our proposed approach, the usage of sensor nodes seems to be almost constant with the increase in parameter matching. In Fig. 3.5, we observe that the effective cost is significantly low in Dec-Safe compared to the traditional Safe-aaS platform, for both static and dynamic approaches. In Safe-aaS, the authors analyzed the cost involved among the various actors. The possible reason behind this is the mapping of the requested decision variables with the generated decisions, in the case of the static approach. On the other hand, in the dynamic approach, we cluster similar decision variables by applying the maximum likelihood function such that the computation time required is minimized. We compare our algorithm with three widely applicable clustering methods – K-Means [91], Mini Batch K-Means [92], and COOLCAT [93] clustering mechanisms. Fig. 3.6 illustrates the performance of four algorithms in the presence of 3000 samples. The training time of Dec-safe is much lesser compared to the other two clustering algorithms, though the centroids are different. Further, the computation is much less in the case of Dec-Safe. We generate the random probability of the decision variables to find the likelihood estimator and thereafter, we cluster these parameters based on the number of decisions. We generate a Silhouette score for each of the clustering algorithms with our samples. It generates scores 40%, 39%, 42%, and 28% in K-Means, Mini Batch K-Means, Dec-Safe, and COOLCAT [93] respectively.

### 3.3.3 Result Analysis

We analyze the performance of the proposed static and dynamic approach for the delivery of safety services to the users under various performance metrics, which are:

#### Static Approach

We interpret and discuss the results obtained using the static approach in terms of parameter matching, effective storage, and total average cost.

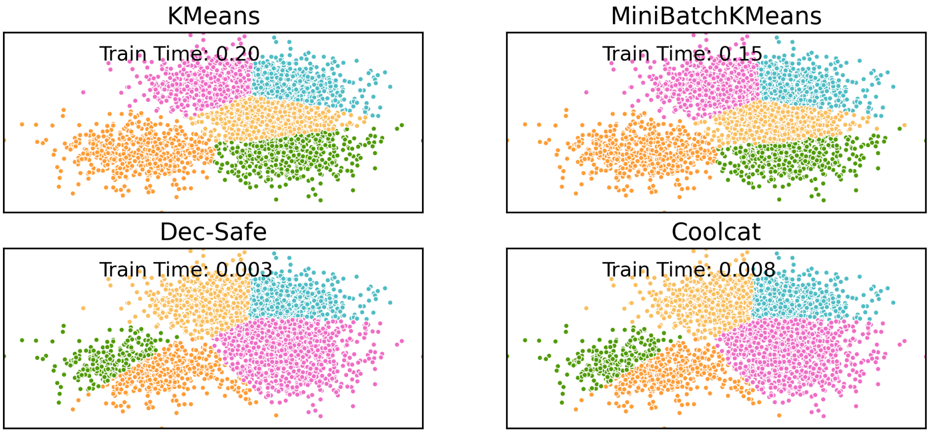


FIGURE 3.6: Comparison among Different Clustering Algorithms

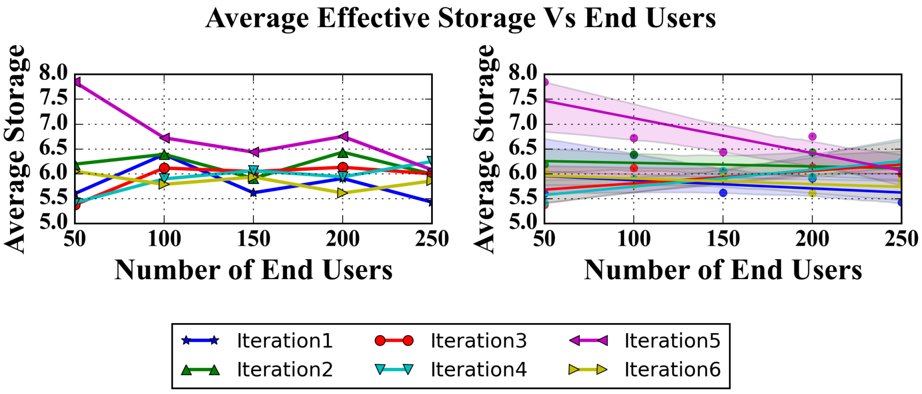


FIGURE 3.7: Effective Storage Usage

*Parameter Matching*: In the proposed static approach, the utility of the $j^{th}$ user is computed based on the value of effective parameter matching and memory space required to generate the decision. We aim to increase the parameter matching, such that the utility of the users is maximized for an optimal value of the memory space required. In Fig. 3.7, we observe that with the increase in the number of users, the average storage space required reduces. This is because the matching between the decision variables is maximized. Additionally, we perform a regression analysis of the fluctuations in the memory space with the number of users.

*Effective storage*: Fig. 3.7 illustrates the variations in the effective memory space without maximizing parameter matching by retaining the number of users from 50-250 along the x-axis. The value of effective memory space initially increases. However, as the number of iterations increases, the value of average memory space decreases by 20-30%. The reason behind such a trend is that as the number of users increases, the probability of matching the decision variables also rises. As a result, the utility of the users is minimized and the memory space required

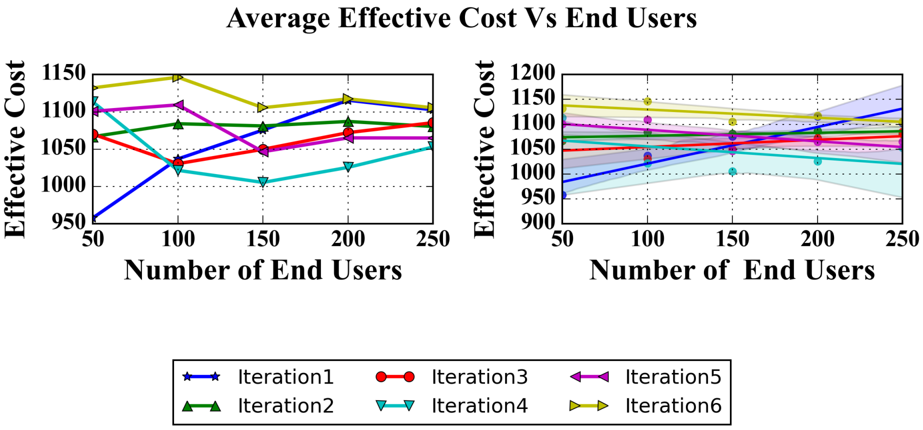


FIGURE 3.8: Cost Analysis

is reduced. However, as the decision parameter matching is not maximized, so we observe a random trend in the memory space.

*Total Average Cost*: Fig. 3.8 illustrates the variations in the average effective cost with the iterations. In iteration 1, as the number of users increases, the average effective cost initially rises. However, we observe that the value of effective cost attains an optimum value after 150 users. The rate of increase in the effective cost reduces as the number of users increases. Further, the parameter matching value increases, and the memory space required to generate the decision minimizes. Therefore, the price charged by the users decreases. We observe a decreasing trend in the regression analysis with the increase in the number of iterations. On the right side of Fig. 3.8, we noticed that after 200 users, the average cost attains a stable value. *Overall Utility*: Fig. 3.9 demonstrates the variations in the overall utility of the users when they select the static approach. We observe that with the increase in the number of users from 50-250, the utilization of the sensor nodes decreases. The possible reason behind this is that the chances of overlap of decision parameters requested by users increase. As a result, the probability of parameter matching increases. Therefore, the effective memory space utilized to generate the decision decreases. On the other hand, with the increase in the number of available decision parameters, the sensor usage initially increases. However, we observe that with the increase in the number of iterations, the sensor nodes utilized seems to follow a decreasing pattern. Further, the overall utility also attains an optimal value with the increase in the number of decision parameters.

### Dynamic Approach

In the dynamic approach, we introduce the concept of a clustering mechanism to generate the decisions for similar decision variables selected by the users. The final decision is obtained from

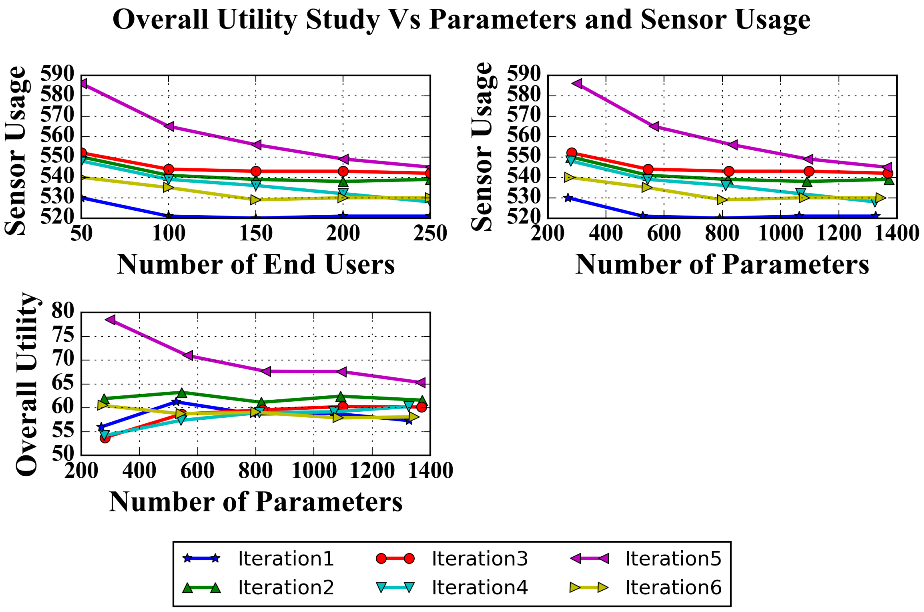


Figure 3.9: Static approach: Overall Utility

the likelihood of similar decision variables. Depending upon the error function, we fabricate the clusters. Our primary aim is to minimize the error between the probability of decision generation and the joint probability of the decision variables.

Decision clustering and Decision generation When registered users require any safety-related information, they select some decision variables. Typically, the decision is generated depending upon their selected decision variables. In this paper, we present a decision-generation mechanism. The proposed clustering technique works on the similarity measure of the selected parameters to provide same decision simultaneously to multiple users. Based on the concept of decision virtualization, the users are grouped to receive the same or similar type of decisions at any time instant in a dynamic environment. In Fig. 3.10, the first figure describes the population of total selected parameters, which are random in nature, and the clustering of parameters in five types of decision by estimating their maximum likelihood among similar parameters. Fig. 3.10 demonstrates the formation of clusters for decision generation with the increase in the number of users. Each time any user modifies the decision variables requested by them, the clustering process is undergone and a decision is generated. Fig. 3.10 illustrates the formation of clusters with the increase in the error. The utility per user is analyzed based on the following parameters -

*Overall Utility*: Fig. 3.11 demonstrate the variations in the utility of the users, who have selected a dynamic approach, with the increase in the number of users. We observe that there exists a decreasing trend in the value of average utility in Fig. 3.11. However, with the increase in the

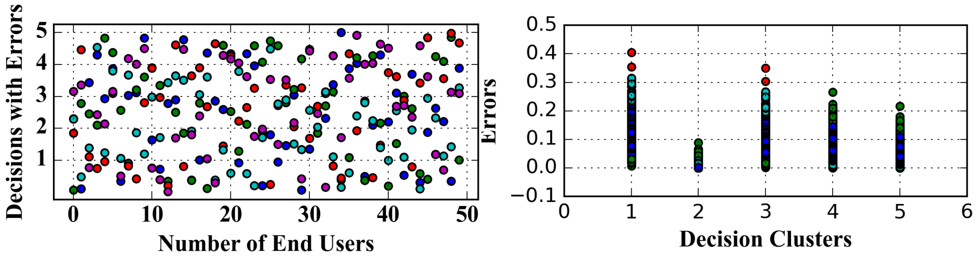


FIGURE 3.10: Clustering of Decisions

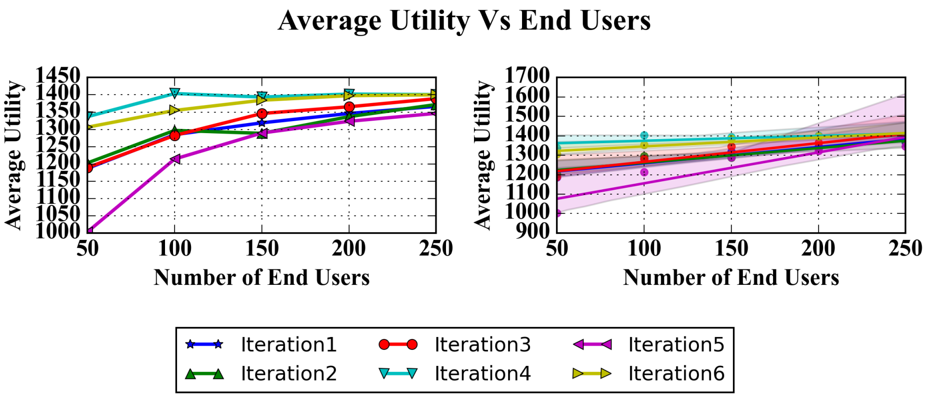


FIGURE 3.11: Dynamic Approach: Effective Utility Analysis

iterations, we observe that the rate of decrease in the value of overall utility is almost constant. Further, regression analysis of the utility illustrates a similar decreasing trend with the increase in the number of users. The possible reason behind such a trend is that with the increase in the number of users, the number of decision parameters requested by them increases. However, we apply a clustering mechanism to group similar decision parameters, which reduces the number of sensor nodes utilized for decision generation and response time delay. As a result, the utility initially decreases steeply and attains an optimal value with the increase in the number of users.

*Optimized Utility*: Fig. 3.12 illustrates the value of utility attained with the rise in the number of users from 50-250. We compute the average utility as per Equation 3.15 and find the minimum value for the optimal response time delay and effective residual energy. The average utility decreases with the number of iterations. However, we observe that for all the iterations, average utility attains a stable value after a certain number of users. Similarly, the regression analysis of the average utility also demonstrates the same increasing trend.
*Average Effective Energy*: Fig. 3.13 demonstrates the effective energy utilized with the variations in the number of iterations. The effective energy initially follows an increasing trend

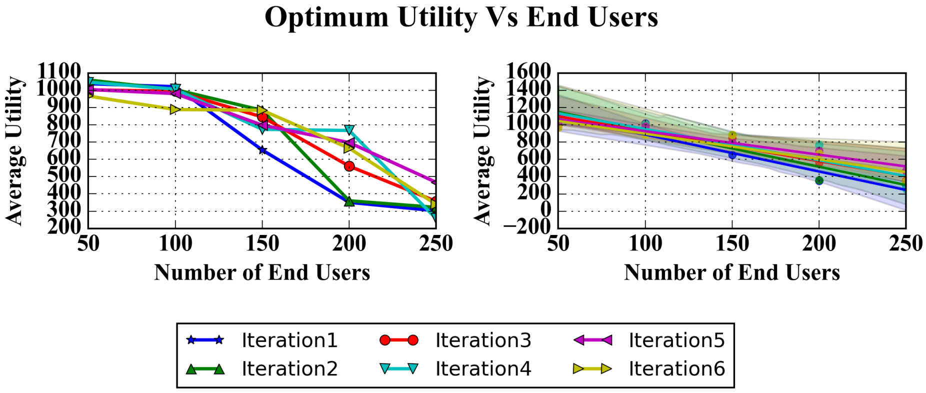


FIGURE 3.12: Utility

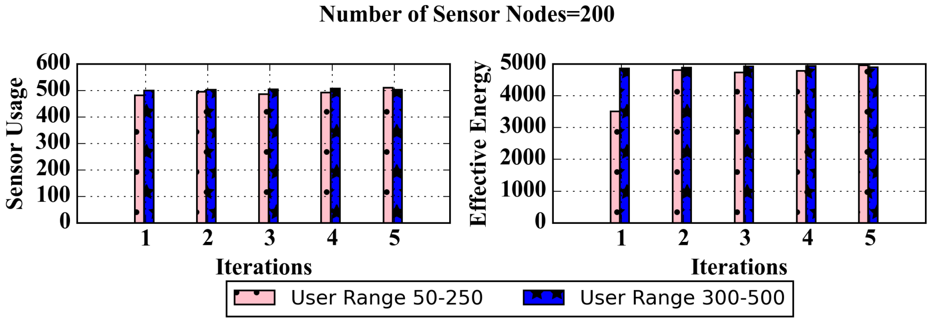
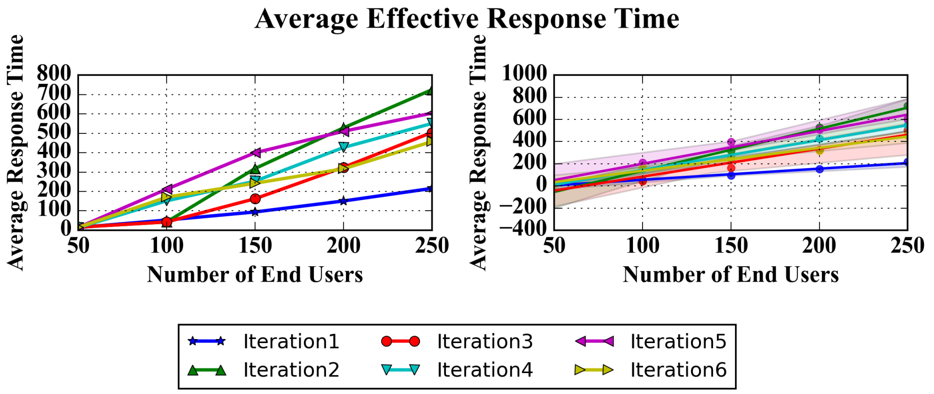


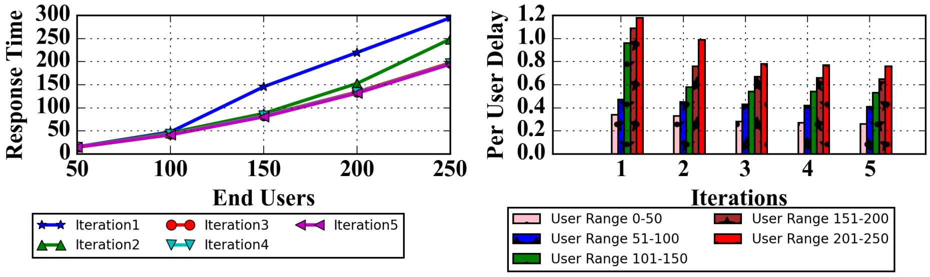
FIGURE 3.13: Sensor Usage, Effective Energy Analysis for Dynamic Approach

with the number of iterations. However, as the number of users increases, the value of effective energy attains a stable value. The regression analysis of the effective energy illustrates a similar increasing trend. As a result, we conclude that the proposed decision-generation scheme is energy-efficient.

*Average Response Time*: We define response time as the time required from the selection of safety-related parameters by the registered users to the delivery of decisions. In this proposed scheme, we calculate total latency in terms of the response time delay. Fig. 3.14a demonstrates the variations in the average response time from generation to delivery of the decisions. We find an increasing trend in the response time with the increasing number of users. However, the increase in the average response time seems to be constant with the increase in the number of iterations. The regression analysis of the response time also illustrates a similar pattern. This increase in the average response time is due to an increase in the number of requests from registered users. Practically, in a real-time environment, the value of decision variables such as congestion/weather-related information fluctuates with time. As we apply clustering for similar decision variables, the increase in the total response time is quite low. We also detect that the



(a) Effective Response Time Analysis



(b) Response Time Vs Delay Analysis

FIGURE 3.14: Response Time Delay Analysis

number of sensors used to sense data is reduced. In Fig. 3.14b, we represent the delay for each user. The total average response time increases with the number of users. On the other hand, the per-user delay decreases/remains constant with the number of iterations.

*Effective cost*: Fig. 3.15 demonstrates the fluctuations in the average cost of users, in case of dynamic approach. We apply the proposed approach over user ranging from 50-250 for 6 iterations. It is observed that the effective cost initially increases. However, with the increase in the number of users, the effective cost seems to attain a stable value. The regression analysis of the effective cost demonstrates a similar increasing trend at the initial stage. We observe that the effective cost decreases after 150 users. Cash inflow from the user end, in case of static approach is quite lower compared to the dynamic approach. As the similarity among the selected decision parameters is increased, effective cost fluctuates randomly with time. Therefore, the average cost charged from the users decreases.

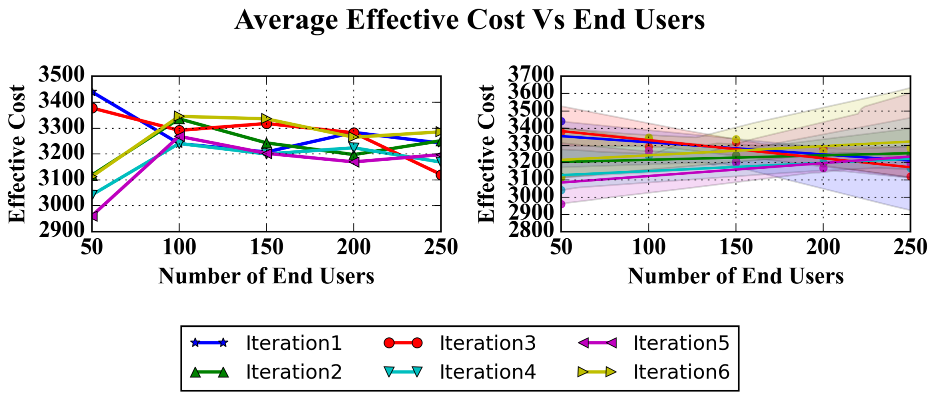


FIGURE 3.15: Effective Cost Analysis

## 3.4 Conclusion

In conclusion, this chapter introduced a dynamic and adaptive framework designed to enhance real-time road safety services within a rapidly evolving transportation ecosystem. By addressing the dual nature of safety-critical decision variables, we provided a comprehensive methodology to ensure a nuanced and context-aware approach to road safety.

Static variables, such as inherent road characteristics like potholes and sharp turns, were addressed using a similarity-based algorithm. This method ensures computational efficiency by generating recommendations with minimal memory usage while maintaining accuracy. Conversely, dynamic variables, which include transient factors like weather conditions and traffic congestion, were tackled using a clustering-based approach. This method leverages user-selected variables to improve decision accuracy by applying the maximum likelihood function to predict and respond effectively to real-time changes.

The proposed mechanism demonstrated superior performance through simulation studies, showcasing a notable reduction in the average effective cost compared to traditional Safe-as-a-Service (Safe-aaS) platforms. By delivering timely and actionable safety insights, this method empowers drivers to make informed decisions, avoiding hazardous conditions, and ensuring safer journeys. The results underscore the potential of adaptive mechanisms to transform road safety dynamics, paving the way for more robust and user-centric transportation solutions. To further improve this safety-service mechanism, we emphasized the importance of accurate decision-making by the SSP. Providing incorrect or partially correct information could lead to accidents. Future efforts will focus on implementing a task distribution approach within the decision layer to enhance performance and accuracy.

# 4

# QoS-Aware Pricing

In the previous chapter, we describe flexibility in safety-decision generation mechanism as per the requirements of end users in Safe-aaS platform. The primary focus of this chapter is to develop a QoS-aware pricing scheme with the flexibility of choosing decision parameters based on the on-road situation for the users. The advancement of the IoT in road transportation helps to reduce road accidents and other hazardous situations by ADAS. Safe-aaS is one of the unique platforms in which users receive safety-related information in real-time on-road hazardous situations. The end users enter the source and destination locations, choose certain number of decision parameters from the suggested ones and make payments through web portal in Safe-aaS platform. The end users receive the decision based on the parameters they have chosen as per their requirements. Safe-aaS is a new platform that offers personalized safety services dynamically to end-users. Decision virtualization allows for numerous end-users to receive generated decisions at the same time. Researchers provide pricing schemes such as static [94], and dynamic [95] pricing schemes, resource utilization-based pricing mechanisms for cloud [96–98], and pricing schemes for sensor cloud platform [99–101]. Tetta *et al.* [102] designed a dynamic pricing scheme to optimally manage the traffic. Various pricing schemes were designed by the researchers for WSNs applied cloud services [97,103], at the network edge for low latency applications [104], and dynamic pricing in mobile social network [105]. Guijarro *et al.* [97] designed a platform, which acts as a broker between the human users and the WSNs. The researchers addressed various problems related to pricing in the domain of WSN and cloud platforms such as profit maximization of users and service providers [97], maintenance of QoS parameters [101], resource allocation [104], dynamic pricing for revenue maximization [105], and performance-based pricing [106]. In cloud computing, the service providers primarily provide preservation of available resources and on-demand plans to the consumers. Ardagna *et al.* [107] proposed two solution approaches for provisioning services in the form of generalized Nash game and proved the existence of equilibrium. Further, there exists heterogeneous types of sensor nodes and multiple sensor owners in a sensor cloud platform. Considering such an oligopolistic market scenario, Chakraborty *et al.* [99] proposed a dynamic pricing scheme to impose trust among the sensor owners to maintain the QoS requirements for provisioning Se-aaS services. Similarly, Roy *et al.* [108] proposed a pricing scheme for provisioning mobile Sensors-as-a-Service (mSe-aaS) such that the profit is optimally distributed among the different actors. We provide a summary of fog/cloud-based pricing schemes in the table 4.1 for road transportation. However, none of these above pricing schemes provide customized prices to the end-users.

Table 4.1: Summary of The Existing Research Works on Road Safety and Pricing

| **Applications** | **IoT** | **Pricing** | **Fog/cloud** | **QoS** | **Latency** |
|---|---|---|---|---|---|
| Safe-aaS [19], [83] | ✓ | D | ✓ | × | ✓ |
| mSe-aaS [108] | ✓ | D | ✓ | ✓ | ✓ |
| Se-aaS [109], [101] | ✓ | D | ✓ | × | × |
| Smart road pricing [110], [102] | ✓ | D | ✓ | × | × |
| Resource allocation [104] | ✓ | D | ×(edge) | × | ✓ |

[Legend: Pricing - Static (S)/ Dynamic (D)]

In this chapter, we develop a QoS-aware pricing scheme to provide end users with personalized, virtualized choices in road transportation in the Safe-aaS application scenario. In this flexible pricing scheme, all the decision parameters are categorized into two categories- low-cost and high-cost parameters.

Generally, in a road transportation environment, low-cost parameters are those that remain unchanged over time, such as the number and depth of potholes, manhole locations, and sharp turns on the road. On the other hand, certain decision parameters such as weather, and road conditions, change their state with time and are set as high-cost parameters. SSPs may tend to suggest high-cost parameters to increase their profit. In a similar way, the end users try to minimize the price to be paid for the service usage to the SSP. This pricing scheme balances the profit earned by the SSPs and the price to be paid by the end users. Several research works focused on the real-time assistance system to minimize road accidents, traffic forecast and evaluate traffic measures by providing alert messages. Safe-aaS provides a common platform where customized safety-related decisions are virtualized to the registered end users as per their requirements. As Safe-aaS is designed for transportation industries, various types of business entities such as sensor vendors, SSPs, end users, and vehicle owners are actively involved in this. End users select decision parameters based on their requirements and pay for them. SSPs make a profit from this payment after paying sensor vendors or other SSPs. There are complex monetary transactions established among actors. Sometimes SSPs may demand higher prices for the service provision from the users. Q-Safe pricing scheme satisfies SSPs and end users as well. In the Q-Safe pricing scheme, we solve the trade-off between minimum price charges from the end users for service usage and quality service provision in terms of correct and accurate decision-making by the SSPs. The overall contribution of this scheme is based on the following issues-

- We provide a flexible pricing scheme for the end users. Price is determined based on the suggested and selected parameters by SSP and end users respectively. We categorize decision parameters as low-cost and high-cost parameters. SSP suggests parameters based on geographical locations and dynamic situations. The final total cost is decided on the optimal number of parameters and type of the parameters.

- We analyze the scenario in which multiple SSPs and users access the service at the same time. We use the Non-Cooperative Multiple-Leader-Multiple-Follower Stackelberg game-theoretic framework to analyze the relationships among SSPs and end-users. SSPs interact as leaders with the end-users as followers. Additionally, we establish the existence of Stackelberg equilibrium in our scenario.

- In safe-aaS, end users tend to reduce the charging price to avail of the service. Similarly, SSPs intend to earn more profit from users. To satisfy SSPs and end-users, we optimize the entire cost. We use the Lagrangian function and Karush-Kuhn-Tucker (KKT) conditions to determine the ideal number of low-cost and high-cost decision parameters that minimize the price charged by SSPs.

- We evaluate and analyze the proposed scheme, Q-Safe, in Python considering various metrics. Extensive analysis results of our proposed scheme proved to be beneficial in terms of the average profit of the service provider and the utility of the end-user, compared to the Per-Subscriber model [97], RegPrice [77], and Prime [108].

## 4.1 Q-Safe: The System Architecture

We consider ITS-enabled road transportation as the application area of Safe-aaS. Safe-aaS platform with service-oriented architecture comprises of five layers-device layer, edge layer, decision layer, decision virtualization layer and application layer. The device layer of Safe-aaS contains heterogeneous static and mobile sensors. The locations of Static sensors are fixed where locations of mobile sensors are changed as vehicles move from one location to another location. The sensed data is primarily processed at the edge layer based on the time-sensitivity.

The pre-processed sensed data is transmitted to the decision layer to generate safety-related decisions. In the decision virtualization layer, logical mapping between decision parameters requested by the users and the decisions generated is established. Thereafter, generated decisions are virtualized to multiple users through virtualization layer. In the application layer, end users select decision parameters after logging into the system and make payments for service usage through the web portal.

There are various business entities such as end users, vehicle owners, SSPs, and sensor owners are involved in this Safe-aaS platform. Complex monetary transactions take place among them. Sensor and vehicle owners rent out their sensor nodes for an amount determined by the SSPs. Further, end users acces safety services on a pay-per-use basis. As a result, the SSP's profit is determined by the remaining amount after payments for the rent from sensor and vehicle owners from the end users payments. SSPs try to maximize their profits, whereas the end users prefer to receive these services at lower prices. However, as safety-related decisions are delivered to end users, it is crucial to maintain the Quality of Service (QoS) of the decisions they receive. In this scheme, Q-Safe, SSPs recommend decision parameters to end users based on their geographic location. We categorize the chosen parameters as *low-price* and *high-price*. Low-price parameters are those whose values do not fluctuate frequently over time. However, the values of the *high-price* parameters change with time. The price is charged to end-users based on the decision parameters provided by them and alterations made after considering SSPs' suggestions, as shown in Fig. 4.1. We present Q-Safe, a QoS-aware pricing strategy, to reduce end-user total costs by selecting the appropriate high- and low-price decision parameters.

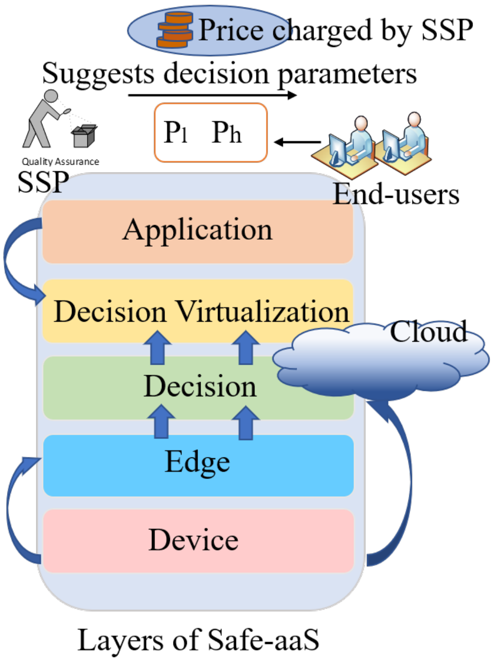


FIGURE 4.1: Q-Safe: The System Architecture

## 4.2 Problem Definition

Let $\mathbb{E} = \{e_1, e_2, \cdots, e_n\}$ be the set of $n$ registered end-users of the Safe-aaS platform. These registered end-users select certain decision parameters from the set $\mathbb{P}$, where $\mathbb{P} = \{p_1, p_2, \cdots, p_m\}$. On the other hand, $\mathbb{N}_s$ represents the set of heterogeneous sensor nodes present in the device layer of Safe-aaS. As discussed in Section 4.1, the SSPs set the price for each of these decision parameters, based on the fluctuation in the value of these parameters with time. Further, the SSPs maintain a mapping between the decision parameters and the price. We characterize this mapping as the one-to-one map. Each of the decision parameters possesses a unique price. On the other hand, based on the price of the decision parameters set by the SSPs, we classify these parameters as – (i) $\mathbb{P}_l$ - the set of decision parameters with low-price, and (ii) $\mathbb{P}_h$ - the set of decision parameters with high-price. $\mathbb{C}_h$ is the cost of a high-price decision parameter, $\mathbb{P}_h^i$. Similarly, $\mathbb{C}_{low}$ be the cost of a low-price decision parameter, $\mathbb{P}_l^i$. The proposed pricing scheme, Q-Safe, has two perspectives – (a) End-user/customer and (b) Safety Service Provider.

*End-user's perspective*: Suppose, the number of parameters selected by an end-user is $\mathbb{P}_s$. The properties of these selected parameters are characterized as follows:

**Property 1.** *$\mathbb{P}_l \subset \mathbb{P}$ and $\mathbb{P}_h \subset \mathbb{P}$. Therefore, $\mathbb{P} = \mathbb{P}_l \cup \mathbb{P}_h$, where $\mathbb{P}_l \cap \mathbb{P}_h$ is not possible, as there are no common parameters between these two sets.*

**Property 2.** *The cost of low-cost decision parameters are always lower than the cost of high-price decision parameters, such that $\mathbb{C}_{low} < \mathbb{C}_{high}$, and $(\mathbb{C}_{low}, \mathbb{C}_{high}) > 0$.*

Let $\mathcal{L}_a^i$ and $\mathcal{H}_b^i$ are the number of low- and high-cost decision parameters selected by the end-users from the set $\mathbb{P}_l$ and $\mathbb{P}_h$, respectively. Therefore, the total number of parameters selected by the $i^{th}$ end-user is

$$\mathbb{P}_s^i = \mathcal{L}_a^i + \mathcal{H}_b^i \tag{4.1}$$

where $\mathcal{L}_a^i$ and $\mathcal{H}_b^i$ are the number of parameters from the set $\mathbb{P}_l$ and $\mathbb{P}_h$, respectively.

*Safety Service Provider's perspective*: The SSPs are responsible for the estimation of the final price for the end-users and check the utility of the available resources. The decision is generated from the data sensed by heterogeneous sensor nodes. We consider $\mathbb{N}_l$ as the number of sensor nodes involved to provide low-price decision parameters and $\mathbb{N}_h$ for the high-price decision parameters. Therefore, the total number of sensor nodes utilized for low-cost and high-cost decision parameter depend upon the selected decision parameters is described as,

$$\mathcal{N}_{tot} = \mathcal{L}_a^i \times \mathbb{N}_{la} + \mathcal{H}_b^i \times \mathbb{N}_{hb} \tag{4.2}$$

The SSP maintains a mapping between the decision parameters and price in $\mathbb{M}_{pp}$.

$$\mathbb{M}_{pp}[i][j] = \begin{cases} \mathbb{C}_{low}, & \text{parameter of } \mathbb{P}_l \\ \mathbb{C}_{high}, & \text{parameter of } \mathbb{P}_h \end{cases} \tag{4.3}$$

The sensor nodes in the device layer are mapped with the decision parameters using their unique identification number. This mapping is maintained by the SSPs, which is mathematically expressed as,

$$\mathbb{M}_{sp}[i][j] = \begin{cases} 1, & \text{if } p_i \text{ is generated from } \mathbb{N}_{sum_j} \\ 0, & \text{otherwise} \end{cases} \tag{4.4}$$

*Centralized Service Utility*: In this proposed approach, we aim to satisfy both the SSPs and the end-users, such that both are benefited. The SSPs deliver the generated decisions to the end-users, and price is charged from them. Based on the number of active sensor nodes at any time instant, the effective energy of the $j^{th}$ sensor node is estimated as,

$$\varepsilon_j^{eff} = \frac{\varepsilon_j^{resi} - (\varepsilon_j^{sense} + \varepsilon_j^{trans})}{\varepsilon_j^{init}} \tag{4.5}$$

where $\varepsilon_j^{resi}$, $\varepsilon_j^{sense}$, $\varepsilon_j^{trans}$, and $\varepsilon_j^{init}$ represent the residual energy, energy consumed for sensing, energy required for transmission, and the initial energy of the $j^{th}$ sensor node at any time instant. Therefore, effective energy consumed to deliver safety services to the $i^{th}$ end-user for $n$ sensor nodes being utilized to generate the decision is $\varepsilon_i^{eff} = \sum_{j=1}^{n} \varepsilon_{i,j}^{eff}$. Further, to provide real-time safety services, any delay may result in a hazardous situation.

**Algorithm 3** Q-Safe: Price Charged from End-Users

```
INPUT: ⟨source, destination⟩ = $\langle S_i, d_i \rangle$, $P_l$, $P_h$, $\mathcal{L}_a^i$, $\mathcal{H}_b^i$.
OUTPUT: Price charged from $i^{th}$ end-user.
PROCEDURE:
 1: for $i = 1$ to $n$ do                                   ▷ $n$: Number of end-users
 2:     for $j = 1$ to $k$ do    ▷ $k$: Number of decision parameters displayed in the Web portal
 3:         $i^{th}$ end-user selects decision parameters
 4:         $\mathbb{P}_s^i$ is computed
 5:         Estimate price charged from $i^{th}$ end-user and his/er utility
 6:         while time = $\tau$ do                          ▷ $\tau$: short time duration
 7:             SSP suggests $j$ parameters to $i^{th}$ end-user, based on $\langle S_i, d_i \rangle$
 8:             if $i^{th}$ end-user agrees then
 9:                 Price and Utility is estimated
10:                 Decision generated
11:             else
12:                 Go to Step 5
13:             end if
14:         end while
15:     end for
16: end for
```

Algorithm 4 provides an overview of the minimum price charged from the end-users. Steps 3 – 5 computes the price charged by the end-users, as per the decision parameters selected. In Steps 6 – 14, a periodic time is used for price re-evaluation. During this period, the minimum price charged from the end-users is estimated by incorporating the decision parameters suggested by the SSPs, until the optimal number of high-price and low-price decision parameters is computed.

The effective time required for this whole process of computation of price charged from the end-users is mathematically represented as,

$$\mathbb{T}_i^{eff} = \frac{\mathcal{N}_{p,i} \times \mathbb{T}_{fixed,i} + \mathbb{T}_{eval,i} + \mathbb{T}_{r,i}}{\mathbb{T}_{comp,i}} \tag{4.6}$$

where $\mathcal{N}_{p,i}$ is the number of times re-evaluation request is processed. $\mathbb{T}_{fixed,i}$ is the fixed amount of time required for each time of re-evaluation. $\mathbb{T}_{eval,i}$ is the utility evaluation time for each period, $\mathbb{T}_{r,i}$ is the response time, and $\mathbb{T}_{comp,i}$ represents the total computation time. Therefore, the utility of the safety service to be provided to the $i^{th}$ end-user is represented as,

$$\mathcal{U}_i = \mathcal{N}_{tot}^i \times \left(\lambda_1 \times \varepsilon_i^{eff} + \frac{\lambda_2}{\mathbb{T}_i^{eff}}\right) \tag{4.7}$$

where $\lambda_1$ and $\lambda_2$ represent the constants for the rate of change of effective energy of the sensor nodes and effective time, such that $1 > (\lambda_1, \lambda_2) > 0$.

### 4.2.1 Pricing Strategies

The end-users and the SSPs interact among them and agrees to the pricing scheme when both are satisfied. The end-users have the intention to select the low-price parameters, such that the price charged by the SSPs is minimized. On the other hand, the SSPs may tend to increase their profit by suggesting high-price decision parameters to the end-users. Therefore, the price charged from end-users must satisfy both the end-users and the SSPs. Considering the above scenario, we design three cases which are discussed as follows.

- *Case 1:* When an end-user selects all low-price parameters, the end-user have to compromise with the real-time safety service, however they have the option to select other parameters. In such a case, the initial price charged by the SSP is represented as,

  $$C_{init} = \mathcal{L}_a \times \mathbb{C}_{low} + \mathbb{C}_{opt} + \mathbb{C}_p \tag{4.8}$$

  where $\mathbb{C}_{opt}$ represents the optional cost of other decision parameters selected by the end-users and $\mathbb{C}_p$ is the processing cost.

- *Case 2:* When an end-user selects mixed parameters – both low- and high-price parameters. This situation provides average quality of service to the end-users within affordable price. In this case, the initial price charged by the SSP is,

  $$C_{init} = \mathcal{L}_a \times \mathbb{C}_{low} + \mathcal{H}_b \times \mathbb{C}_{high} + \mathbb{C}_p \tag{4.9}$$

- *Case 3:* When an end-user selects only high-cost parameters, the initial price charged by the SSP is mathematically represented as,

$$C_{init} = \mathcal{H}_b \times \mathbb{C}_{high} + \mathbb{C}_{opt} + \mathbb{C}_p \tag{4.10}$$

After all the above cases, we represent the total price charged by any SSP from an end-user for these above-mentioned cases as follows –

$$C_{total} = \sum_{j=1}^{\mathcal{N}_p} (C^j_{init} + C^j_r) \tag{4.11}$$

where $C^i_r$ is the re-evaluation cost for $\mathcal{N}_p$ re-evaluation requests. To maintain the quality of service (QoS) provided to the end-users, satisfy the utility of service provided to them, and select the appropriate number of high- and low-price decision parameters is an essential aspect of concern. Therefore, a trade-off is to be maintained between the satisfaction of the end-users and the price charged by the SSPs.

$$C^i_{total} = \beta \times C^i_{init} \times \mathcal{U}_i + C^i_r \tag{4.12}$$

### 4.2.2 Quality of Service (QoS)

Typically, in Safe-aaS, depending upon the decision parameters selected, the end-users make payment. In Q-Safe, the SSPs suggest to the end-users certain decision parameters. Further, QoS depends on the efficiency of the heterogeneous sensor nodes. In a recent research work, Roy *et al.* [108] proposed an optimal pricing scheme considering the quality of service. The authors designed QoS in terms of the efficiency of sensor nodes. Further, the service return of the service provider is measured in terms of the type of end-users and time factor. Motivated by this concept, we mathematically define the efficiency, $\mathbb{E}^i$, and quality of service, $Q^i$ for $i^{th}$ end user as,

$$\mathbb{E}^i = \frac{\varepsilon^{eff}_j \times (\mathbb{T}_{c,j} + \mathbb{T}_{t,j})}{\mathbb{T}_{r,i}} \tag{4.13}$$

Therefore, $Q^i$ is represented as,

$$Q^i = \alpha \times \mathbb{E}^i \times \mathcal{U}_i \tag{4.14}$$

where $\mathbb{T}_{c,i}$ and $\mathbb{T}_{t,i}$ are the time required to collect and transmit data from edge layer by $j^{th}$ sensor node for $i^{th}$ end user. $\mathbb{T}_{r,i}$ is taken to response to $i^{th}$ end user.

### 4.2.3 Game Formulation

In Safe-aaS, the customers register to the platform, select certain parameters and make payment through the Web portal. A decision is delivered to them. In our proposed pricing scheme, we introduce a suggestive method using which the end-users select their decision parameters. The SSPs suggest certain decision parameters to the customers, depending upon their selected source and destination details. To map the strategic interactions among the end-users and the SSPs, we apply *Non-Cooperative Multiple Leaders Multiple Followers Stackelberg* game-theoretic approach. The SSPs act as leaders and the end-users act as followers. Suppose, $\mathbb{E}^x$, such that $\mathbb{E}^x \subset \mathbb{E}$ and $(1 \leq x \leq n)$, set of end-users which act according to the pricing scheme declared by the SSPs, $\mathbb{Z}^y, 1 \leq y \leq q$.

*Non-Cooperative Stackelberg Game - The Justification*: The end-users first select certain decision parameters, among the ones displayed in the Web portal. Thereafter, based on their source and destination details given by the end-users and to maintain the Quality of Service (QoS), the SSPs suggest certain decision parameters. The price charged from the end-users and their utility is estimated during each re-evaluation. The SSPs possess the intention to increase their profit as well as satisfy the end-users with the price charged. Therefore, a dynamic scenario exists, where we map the interactions among the SSPs and the end-users with a non-cooperative game. Each of the players, the leaders and the followers, take their decisions independently in the game. The leaders first put forth their strategies or suggest the decision parameters. Based on their strategies or suggested decision parameters, the followers/end-users select their decision parameters.

**Lemma 1.** *The event of selection of the decision parameters by the end-users and those suggested by the SSPs is a pairwise, dependent event.*

*Proof.* We consider $P^x$ as the decision parameters initially selected by the end-users and $P^y$ as those suggested by the SSPs. We design the selection of the decision parameters as an event. Suppose, the probability of occurrence of these events be denoted as **P**. Therefore,

$$\mathbf{P}(P^y \cap P^x) = \mathbf{P}(P^x)\mathbf{P}(P^y|P^x) \tag{4.15}$$

where $\mathbf{P}(P^y|P^x)$ represents the probability of occurrence of the event $P^y$ when $P^x$ has already occurred. □

Therefore, the strategic form of the game is defined as –

$$\xi^i = (\mathbb{Z}^y \cup \mathbb{E}^x)_{(x \in n, y \in q)}), (\mathcal{S}_L^y, \mathcal{S}_F^i, \mathcal{U}_L^y, \mathcal{U}_F^i)_{(i \in n, y \in q)} \tag{4.16}$$

The various parameters of the game are – (i) $\mathbb{Z}^y$, set of leaders/SSPs,(ii) $\mathbb{E}^x$, set of followers/end-users, (iii)$\mathcal{S}_L^y$, strategies of the leaders, (iv)$\mathcal{S}_F^i$, strategies of the followers, (v) $\mathcal{U}_L^y$, the utility function of the leaders, and (vi) $\mathcal{U}_F^i$, the utility function of the followers.

*Strategies of the leaders*: The leaders put forth their strategies depending upon the different pricing strategies, which are described in Section 4.2.1. Therefore, strategy of the leaders is mathematically represented as, $\mathcal{S}_L^y = \{C_{init}, \mathbb{C}_{low}, \mathbb{C}_{high}, \mathbb{C}_p\}$.

*Strategies of followers*: The followers place their strategies, $\mathcal{S}_F^i$, depending on the type of decision parameters selected by them. Therefore, $\mathcal{S}_F^i = \{\mathcal{L}_a, \mathcal{H}_b\}$.

To satisfy the end-users as well as the SSPs requests, we aim to minimize the total costs, depending on the optimal number of low-cost and high-cost decision parameters selected.

**Theorem 1.** *There exists a unique Stackelberg equilibrium, for the total costs charged by the SSPs from the end-users. To estimate the total cost, we consider a given re-evaluation cost, effective residual energy of the sensor nodes, number of sensor nodes used, and time required for the entire process of decision parameters selected by the end-users and suggestions provided by the SSPs.*

*Proof.* In our proposed pricing scheme, each of the end-users requests certain decision parameters, and decision is provided by the Safe-aaS platform. Further, the SSPs tend to minimize their utility and increase their profit, such that the decision is provided to the end-users, utilizing the minimum number of sensor nodes and their energy consumed, within a bounded time period. Therefore, the optimization function is mathematically represented as,

$$\underset{\mathcal{L}_a^i, \mathcal{H}_b^i}{argmin} \quad C_{total}^i \tag{4.17}$$

subject to, $m \geq (\mathcal{L}_a^i + \mathcal{H}_b^i)$, $C_{high} > C_{low}$, $C_r \geq 0$, $C_p > 0$, $\mathcal{N}_{tot}^i > 0$, and $0 \leq (\varepsilon_i^{eff}, \mathbb{T}_i^{eff}) \leq 1$. The maximum number of decision parameters displayed in the Web portal is represented as $m$. In order to simplify the optimization function, we apply *Lagrangian* function, which is represented as,

$$\begin{aligned} L^i = & -C_{total}^i - \mu_1(\mathcal{L}_a^i + \mathcal{H}_b^i - m) - \mu_2(C_{low} - C_{high}) \\ & - \mu_3(C_p) - \mu_4(C_r) - \mu_5(\mathcal{N}_{tot}^j) - \mu_6(\varepsilon_i^{eff} - 1) \\ & - \mu_7(\mathbb{T}_i^{eff} - 1) \end{aligned} \tag{4.18}$$

where $\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6$, and $\mu_7$ represent the *Lagrangian Multipliers*. Further, to solve Equation 4.18, we use *Karush-Kuhn-Tucker* (KKT) conditions. The *dual feasibility* and *complementary slackness* conditions are represented as follows:

$$\frac{\partial L^i}{\partial \mathcal{L}_a^i} = -(\beta C_{low} + \beta \mathcal{L}_a^i \mathbb{N}_{la}) + \mu_1 = 0 \tag{4.19a}$$

$$\mu_i(X) = 0 \quad \text{and } \mu_i \geq 0, \forall i = \{1, 2, \cdots, 7\} \tag{4.19b}$$

where $X$ represent the constraints of the Equation 4.17. On solving Equation 4.19, we obtain the optimal value of $\mathcal{L}_a^i$. Similarly, we perform the first order derivative of Equation 4.18, with respect to $\mathcal{H}_b^i$ and applied the KKT conditions, to obtain the optimal number of high-cost parameters. Therefore, the optimal value of $\mathcal{L}_a^{i,*}$ and $\mathcal{H}_b^{i,*}$ are represented as:

$$\mathcal{L}_a^{i,*} = \frac{1}{\mathbb{N}_{la}}\left(\frac{\mu_1}{\beta C_{low}} - 1\right) \tag{4.20a}$$

$$\mathcal{H}_b^{i,*} = \frac{1}{\mathbb{N}_{hb}}\left(\frac{\mu_1}{\beta C_{high}} - 1\right) \tag{4.20b}$$

Based on the optimal values of $\mathcal{L}_a^{j,*}$ and $\mathcal{H}_b^{j,*}$, we obtain the minimum total cost charged by the SSP from the end-users. ☐

## 4.3 Performance Evaluation

### 4.3.1 Simulation Design

To evaluate and analyze the performance of the proposed pricing scheme, Q-Safe, we vary the user entities from 0 to 500 and the number of sensor nodes from 200 to 600, in a simulation area of $10 \times 10 km^2$. We randomly deploy the sensor nodes in the simulation region. The various simulation parameters used are listed in Table 4.2.

TABLE 4.2: Simulation Parameters for Q-Safe

| **Parameters** | **Values** |
|---|---|
| $\mathbb{P}_l, \mathbb{P}_h$ | 10 |
| $\mathbb{N}_l, \mathbb{N}_h$ | 200 |
| Decision parameters | 10 |
| Cases for price charged | 3 |
| $C_{low}$ | 100 – 500 |
| $C_{high}$ | 501 – 1000 |
| $C_r$ | 100 |
| $\lambda_1, \lambda_1, \beta$ | 0 – 1 |

### 4.3.2 Benchmark Solution

Existing research works discussed various dynamic pricing schemes to fulfill the demand of both SSP and customers, in terms of profit of SSP and maintain the quality of service (QoS). Guijarro *et al.* [97] proposed a two-sided payment scheme in a service platform, based on WSNs. Their designed platform acts as a mediator between the consumers and the WSNs, where both the service providers as well as consumers post their prices to maximize their profit. On the other hand, Roy *et al.* [108] proposed a dynamic pricing scheme for providing Sensors-as-a-Service (Se-aaS) in the mobile sensor cloud environment. They considered the quality of service provided by the sensor nodes in terms of service return, the price charged by the Sensor Cloud Service Provider (SCSP). Additionally, we compare another recent research work on pricing in the Safe-aaS platform as a benchmark scheme. Considering the type of road in different geographical regions and the presence of homogeneous sensor nodes, Roy *et al.* [77] proposed a region-based pricing scheme. The authors calculated the price charged from the end-users based on fixed cost, variable cost, and maintenance cost. We termed the pricing scheme proposed by Guijarro *et al.* [97], Roy *et al.* [108], and Roy *et al.* [77] as Per-Subscriber Model, PRIME, and RegPrice. However, none of these existing schemes consider the quality of service to be provided to the end-users in terms of their requirement, and suggestion is not given by the service provider.

We analyze the profit of SSP with the increase in the number of end-users, as illustrated in Fig. 4.3. We observe that the average profit in the proposed scheme, Q-Safe, is improved by 70.88%, 52%, and 77% compared to the Per-Subscriber model, PRIME, and Reg-Price in the presence of 200 sensor nodes. We increase the number of end-users from 50-500 along the x-axis. The possible reason behind the increase in the profit of SSPs is the rate of increase in the demand of safety services by the registered end-users. Moreover, these end-users select the optimal number of high- and low-cost decision parameters. We observe that the profit of the SSPs varies randomly with the increase in the number of customers. As the price charged to the customers varies with the number and type of selected parameters by them, therefore, the average profit of the SSPs also fluctuates. Fig. 4.4 demonstrates the variations in the utility of the proposed scheme, Q-Safe with the existing benchmark schemes, Per-Subscriber, PRIME, and RegPrice. We vary the number of end-users from 0 upto 500 with an interval of 50, along the x-axis. Interestingly, we observe that the average utility of Per-Subscriber model decreases with the increase in the number of end-users, whereas the utility of Q-Safe increases in the presence of 200 sensor nodes. We observe that the value of average utility is reduced by 5%, 16%, and 17% with respect to PRIME, Per-Subscriber, and RegPrice. One of the possible reasons behind this trend in the average utility is that the effective time required to generate a decision decreases with the increase in the number of end-users. The possibility of similarities among the decision parameters selected by the end-users increases with the rate of increase

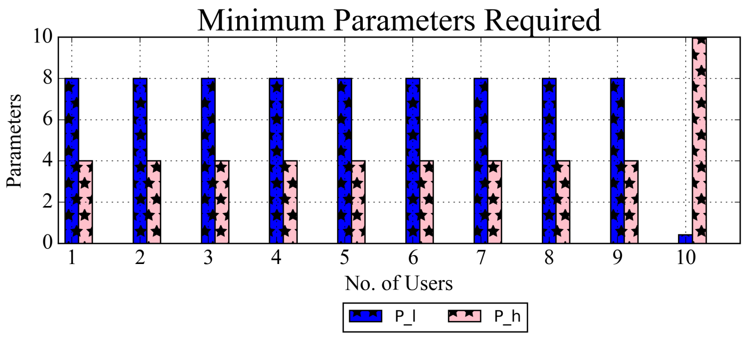


FIGURE 4.2: Optimized Parameters Per User

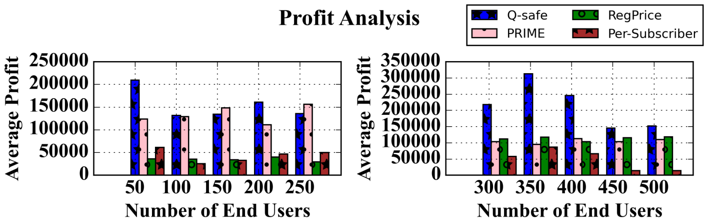


FIGURE 4.3: Profit Analysis

in their number. Therefore, the time required in processing, analysis, and generation of the decision reduces. Fig. 4.5 illustrates the variations in the average cost of end-users in the presence of 200 sensor nodes in the environment. It is studied that the cost or price charged from end-users using our proposed scheme is quite low compared to the other existing schemes. The price charged from the end-users is highest in case of Per-Subscriber model, in contrast to PRIME and RegPrice. The possible reason behind this is that the concept of low- and high-price decision parameters and the suggestion provided by the service provider. Based on the selected type and number of decision parameters, the price is charged from the end-users. Further, the decision is provided to the them accordingly. Fig. 4.6 demonstrates the variations in the QoS of the proposed scheme with other existing schemes, PRIME, RegPrice, and Per-Subscriber model. It is shown that the QoS values follow raising trend with the increase in the number of customers. However, the rate of increment in QoS is comparatively high in Q-Safe with respect to other benchmark mechanisms. One of the possible reasons behind such a trend is that the value of utility and effective energy is high compared to the other existing schemes, as illustrated in Fig. 4.4 and 4.7.

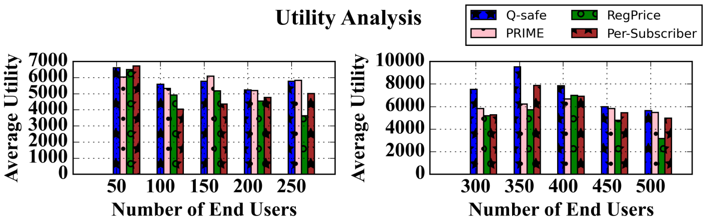


FIGURE 4.4: Utility Analysis

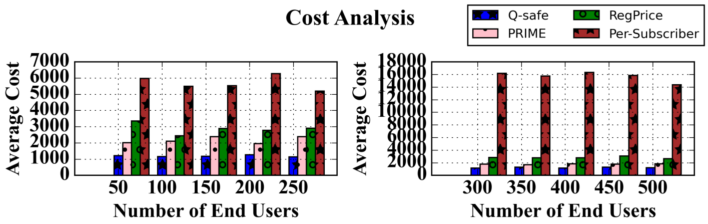


FIGURE 4.5: Cost Analysis

### 4.3.3 Result Analysis

In our proposed approach we describe how price is charged from end users through maintaining the quality of service. To maintain QoS of the safety services, we consider various parameters involved in the proposed pricing scheme such as utility of service, total cost, energy consumed, time, error characteristics, and optimal number of selected parameters by the end-users, to characterize it. In this section, we study and analyze the behavior of these parameters for helping customers by delivering safety services at an optimal cost.

*Effective Energy:* Fig. 4.7 illustrates the variations in the effective energy consumed, with 200 sensor nodes. We vary the number of end-users from 20–200 at an interval of 40 along the x-axis. We observe an increasing trend in the average effective energy, both in case of high- and low-cost decision parameters. The probable reason behind this is that with the increase

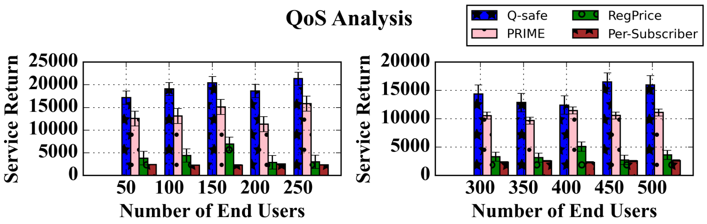


FIGURE 4.6: QoS Analysis

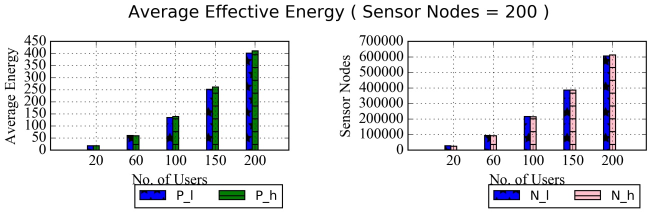


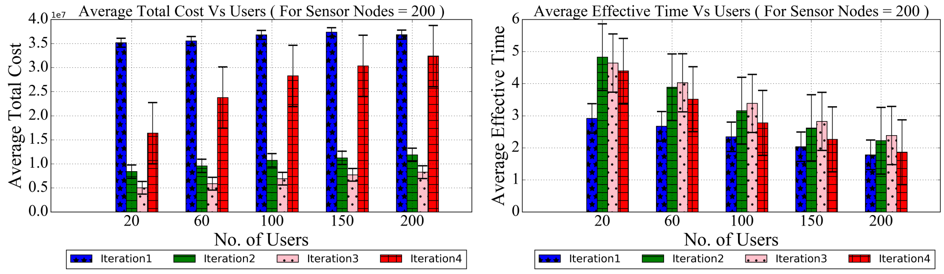
FIGURE 4.7: Average Effective Energy

in the number of end-users, the number of decision parameters (high- and low-cost) selected by them increases. Therefore, the total sensor nodes in the decision generation process also increases. We consider the energy consumed by used sensor nodes for five iterations. As a result, the effective energy consumed increases. In another analysis of Fig. 4.7, we observe that with respect to the increase in the number of end-users, the number of sensor nodes involved in the decision generation correspondingly increases. As a result, the number of sensor nodes required to provide the information of the low- and high-cost decision parameters also increases.

*Total Cost:* Fig. 4.9a demonstrates the average total cost of the decision parameters incurred by an end-user at different iterations during the re-evaluation of the cost before optimization. In our proposed approach, the price is charged from the end-users, based on their selected decision parameters. SSP provides two types of parameters- high- and low-cost. Based on the decision parameters selected by the end-users, the total cost is estimated. We observe that the average total cost follows a decreasing pattern for different iterations and an increasing pattern with the increase in the number of end-users. In Fig. 4.9a, the total average cost after iteration 1 is quite high. This signifies that most end-users select high-cost decision parameters. However, after iteration 4, the average total cost is quite high compared to iteration 3. The possible reason behind this is that the end-users select more high-cost decision parameters than low-cost parameters in iteration 4. As a result, the utility of the service to be provided to the end-users changes. Further, selected parameters vary with the inclusion of decision parameters suggested by the SSPs. Moreover, as the decision parameters recommended by the SSPs are incorporated, the average total cost charged from the end-users decreases, after each re-evaluation.

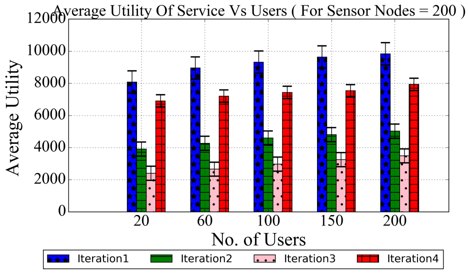
*Effective Time:* Fig. 4.9b illustrates the variations in the average effective time with the increase in the number of end-users. We observe that the effective time decreases with the increase in the number of end-users by 58.12%, in the presence of 200 sensor nodes. Additionally, the effective time decreases with different iterations. The possible reason behind this trend is that the number and type of decision parameters selected by the end-users may overlap. Therefore, the time required to evaluate and generate the decision, and the number of times re-evaluation requests are processed, are minimized.

*Utility:* Fig.4.8c demonstrates the variation in the utility of safety services to be provided



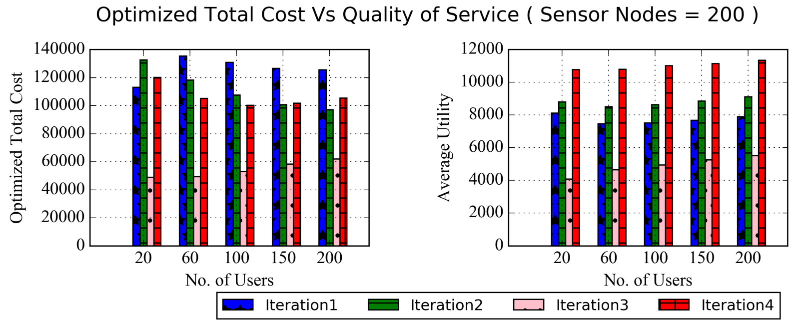
(a) Average Total Cost

(b) Effective Time

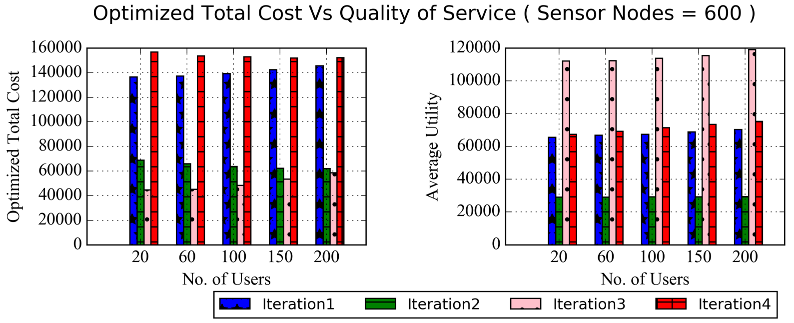


(c) Utility

FIGURE 4.8: Variation in Average Cost, Time, and Utility



(a) a



(b) b

FIGURE 4.9: Variation of Optimum Cost with QoS

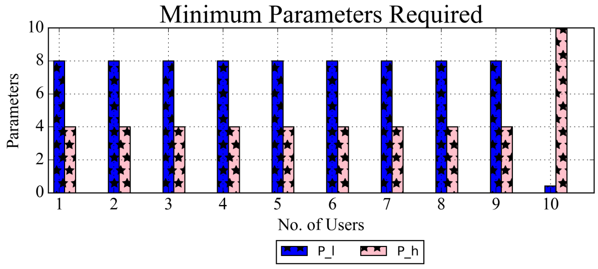


FIGURE 4.10: Optimized Parameters Per User

to the end-users. We observe that with the increase in the number of end-users, the utility of service increases by 34.35%. We estimate the utility of safety service as per Equation 4.7. The utility of the services provided to the end-users increases with the increase in the number of decision parameters selected and decreases with the increase in the time required for decision generation. From Fig. 4.7, and 4.9b, we study that the effective energy consumption increases and the effective time required for estimation of total cost decreases with the increase in the number of end-users. Therefore, as per Equation 4.7, with the increase in the effective energy consumed and decrease in the effective time, the utility of service also increases.

*Optimal number of parameters selected:* Fig. 4.10 depicts one example of the optimal number of high-price and low-price decision parameters selected by the end-users, such that the total price charged by the SSPs is minimized. In Fig. 4.10, we consider the optimum number of high-range and low-range parameters are selected in the presence of 10 end users. We estimate the optimal value of $\mathcal{L}_a^{j,*}$ and $\mathcal{H}_b^{j,*}$ from the solution of the optimization function, as given in Equation 4.20. Therefore, the optimal number of decision parameters selected by the end-users vary. Fig. 4.9 demonstrates the variation of optimal cost and average utility with the increase in the number of end-users from 20 to 200. We observe that both the minimum total cost charged by the SSPs and average utility at different iteration vary randomly in the presence of 200 and 600 sensor nodes.In comparison with Fig. 4.9a, the total cost is significantly minimized with every iteration for 200 sensor nodes.

However, there exists a rising trend with the increase in the number of end users. The possible reason behind this trend is for a similar number of high-range and low-range parameters, the number of end users are increased.

*Error Characterization:* Fig. 4.11 illustrates the characterization of error in the estimation of energy, utility, time, total cost, and the optimal number of high-price and low-price decision parameters selected. We compute the energy consumed based on selected parameters by the customers (error estimated in Energy1 graph) and used sensor nodes in the decision generation (error estimated in Energy2 graph) process. We observe that the error occurred during different iterations is significantly low in case of energy consumed, while the error is quite high in case

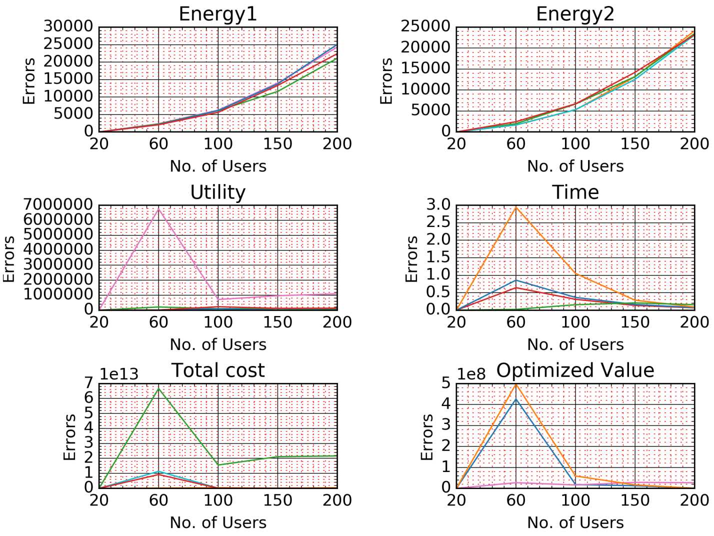


FIGURE 4.11: Error Characterization of Optimization Parameters

of utility, effective time, total cost, and the optimal number of high- and low-cost decision parameters. Error characterization is required to provide the clear concept about the trend of variations in the parameters. We observe the outcome after five iterations and found that the error is minimum for every parameter. However, the total cost varies because of different types of parameters chosen by the customers. From the error characterization graphs, the future behavior of all the decision parameters can be predicted over the number of end-users.

## 4.4 Conclusion

In this chapter, we introduced a QoS-aware dynamic pricing model called Q-Safe, designed to deliver customized safety services to end-users. The Q-Safe framework employed a two-way pricing mechanism that is mutually beneficial for both end-users and SSPs. The model leveraged high-price and low-price decision parameters, displayed on a user-accessible Web portal, to guide pricing decisions. Initially, end-users select certain parameters based on their preferences, after which SSPs can recommend additional decision parameters tailored to the users' geographical location. This approach enabled a dynamic interaction between users and SSPs, which we modeled using a non-cooperative Multiple-Leader-Multiple-Follower Stackelberg game-theoretic framework.

To implement this dynamic pricing scheme, we considered various factors, including the effective energy consumed, the computational time required to calculate the total cost, and the sensor nodes utilized in the decision-making process. These elements were used to construct a utility function representing the quality of service provided to end-users. By analyzing the utility and decision parameters selected by users, we determined the total cost incurred by end-users for the service.

Further, we formulated an optimization function that aims to minimize the overall cost charged to end-users by selecting an optimal mix of high-cost and low-cost decision parameters while ensuring the desired QoS levels. Through this optimization, the Q-Safe model can offer an adaptable and efficient pricing strategy that balances cost and quality for both end-users and service providers.

# 5

# Edge-Intelligence Based Safe-aaS

The recent advancements in the field of ITS are primarily due to the wireless vehicular sensing and communication technologies, which resulted in improved on-road safety of individuals. Fallgren *et al.* [111] examined the 5G technologies required for the convergence of communication among the vehicles with the help of mobile applications. The key objectives of 5G-based communication technologies are to minimize latency, enhance reliability, increase scalability, and secure inter-vehicular communications. On the other hand, the improvement in the traffic efficiency and minimization of on-road congestion are the essential aspects of concern for both safety of vehicles and drivers. Lin *et al.* [112] proposed an edge computing-based public vehicle system for enabling smart transportation. Similarly, Boban *et al.* [113] found the research gap in the V2X communications and proposed a V2X random access network for 5G scenario. Various artificial intelligence tools are used to analyse traffic and accidents to reduce real time hazards. Cafiso *et al*, Vasmari F *et al* [114] proposed fuzzy control system for traffic analysis to reduce road accidents. Ali *et al.* [115], Izquierdo *et al.* [116] studied the characteristics of the drivers, traffic and applied ANN method to study the hazard created by driver's behaviour in the accidents. The authors proposed a mechanism to alert the driver to avoid road accidents. The primary objective of the automation is to reduce vehicle collisions and prevent hazards. Roy *et al.* [19,83,90] developed a unique platform for transferring safety-related decisions to end users. In Safe-aaS, the SSPs manage and maintain the platform in centralized manner. To ensure uninterrupted services for end users, the authors designed a dynamic service handoff mechanism among the SSPs. Ranadheera et al. [117] proposed a distributed mechanism for computation offloading and for activating the appropriate mobile edge computing server. They utilized the theory of minority games to develop a solution for their proposed approach. However, none of the existing schemes address the implementation of an edge server that delivers safety-related information to users with ultra-low latency. As safety-related decisions are provided to the end-users, so any delay in delivery of these decisions is undesirable. Further, another major challenge associated with the Safe-aaS platform is to maintain accuracy of the generated decisions. Edge servers are widely popular for their computational and processing capabilities at the "edge" of the network [118]. Typically, based on their location, the edge servers are categorized as regional, network, and on-premises edge. Unlike traditional methods, the complex analysis and storage of data are done at the user's location. The edge servers are selected with the criteria, which include computation capability constraints and previous performances. Edge severs are placed at a certain geographical location. The overall computational and processing costs are also minimized. Motivated by these facts, we introduce the concept of edge intelligence into the existing Safe-aaS platform for providing accurate decisions to the end users with ultra-low latency.

In this proposed scheme, we intend to introduce the concept of edge server in the Safe-aaS infrastructure. The specific contributions of this work are as follows:

- We present the concept of edge intelligence layer in the Safe-aaS platform, which comprises distributed edge servers. The sensed data is primarily processed at the edge nodes. We apply ANN at the edge nodes to classify the edge servers and fuzzify the decisions at the edge server end, thereby minimizing the computing overload of the cloud server. Decisions are further propagated to decision layer.
- We design an algorithm to select the appropriate edge server for processing the sensed data and decision generation. We estimate the computing density to mark edge servers in the active or inactive state. We show that the computing density is normally distributed.
- We provide unique decision generation mechanism at the edge server with the help of fuzzification. We categorize the decision parameters to generate the decision class in which these parameters belong.
- Lastly, we compute total delay with/without the edge intelligence layer. We observe that with the introduction of the edge intelligence layer, the total delay is minimized to 90.58% compared to the delay without the edge layer.

## 5.1 Necessity of Edge Server in Safe-aaS

In the IoT scenario, the edge servers compute, process, analyze, and store data near the location of the end-users. Typically, in a traditional Safe-aaS platform, the edge layer comprises edge nodes, which primarily process the time-sensitive raw sensed data. The sensed data, which are not time-sensitive, are processed in the cloud. As road-safety information are delivered to the end-users, so any form of delay in delivery may lead to a hazardous situation. Depending upon their type, the edge nodes present in the edge layer possess limited computation and processing capability. In the presence of edge servers at the edge layer, the delay incurred in the processing and delivery of the decisions is significantly reduced. Further, the management of the security of these generated decisions and credentials of the end-users is necessary. The following attributes characterize the edge server-enabled Safe-aaS platform:

***Minimizes latency/delay incurred in decision generation*:** We consider that the edge layer of the Safe-aaS platform consists of distributed edge servers. Because of this distributed nature, the decision is generated at any of the activated servers, hence, the delay incurred is minimized. Further, the generated decisions are stored at these edge servers placed close to the end-users. On the other hand, multiple end-users register and request decision parameters at the same time. Therefore, the decision parameters requested by them may overlap. The corresponding decisions can be cached and stored at the edge servers. Hence, the cache access time is minimized.

***Facilitates computation offloading*:** Each of the edge nodes possesses two modules - communication and computation. In the traditional Safe-aaS platform, the communication module uploads the time-sensitive data from the edge nodes to the cloud, after primarily processing them. Based on the type of these edge nodes, their processing and computation capability differ from each other. Therefore, when the edge nodes offload the task to the cloud, high bandwidth is necessary. Due to limited bandwidth, a delay is incurred in uploading the data and some data packets may be dropped. Further, any drop of data packets may result in the generation of inappropriate/partially correct decisions. However, the distributed edge server-enabled Safe-aaS platform minimizes the network overhead costs, while uploading the data and offloading the tasks.

***Extends storage and processing*:**The edge servers enable storing the generated decisions and the credentials provided by the end-users during registration. Additionally, the information regarding the sensor and vehicle owners, and sensor nodes rented by them to the Safe-aaS platform is stored in these servers. The processing of the sensed data and application of complex analysis techniques is done at the distributed edge servers.

***Eliminates the issues associated with the performance of SSPs*:** A SSP is a centralized entity, who manages and maintains the entire Safe-aaS platform. The SSPs pay an amount as a penalty in case of delay in delivery of the generated decisions to the end-users [**?**]. The integration of the edge server with the Safe-aaS platform helps to achieve ultra-low latency. Therefore, the profit of SSPs increases because the penalty cost reduces. As a result, the performance of SSPs as well as end-users satisfaction, is improved.

***Imposes security on the generated decisions*:** The raw sensed data are directly processed at the edge servers and decisions are generated. Therefore, any form of external malware attacks on data can be avoided and the security of the generated decision is managed. Further, the credentials of end-users, sensor owners, and vehicle owners are stored in the distributed edge servers, without permitting any third party to access these data. Hence, the security of the decisions as well as information of end-users, sensor nodes rented, and their owners is preserved.

## 5.2 Distributed Edge Server-Enabled Safe-aaS Infrastructure

In Safe-aaS platform, registered end users select certain decision parameters among the ones available and pay the service charge through the Web portal. End users get benefited by the safety decisions made from the selected parameters. The end-users possess no knowledge regarding the sensor nodes and the decision generation process. To minimize the delay incurred in decision generation, improve computation offloading, upgrade the end-users experience, and increase the profit of SSPs, we introduce the concept of edge Server into the Safe-aaS platform.

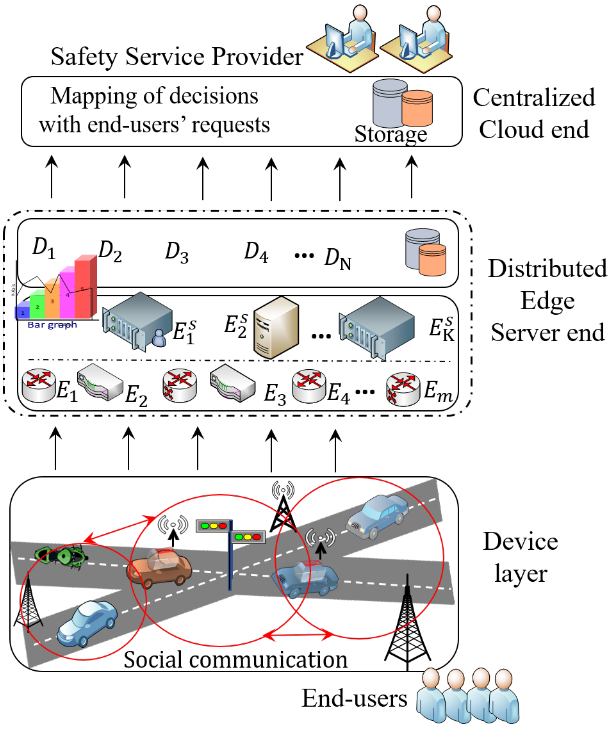


FIGURE 5.1: Edge Server-Enabled Safe-aaS

We consider that the edge layer of Safe-aaS platform is replaced by the distributed edge server-enabled layer. The SSPs are responsible for the management and maintenance of the edge servers. The various layers of the ***Distributed Edge Server-Enabled Safe-aaS*** infrastructure are as follows:

***Device Layer*:** This layer comprises heterogeneous types of static and mobile sensor nodes, which are either deployed at a geographical location or placed into the vehicles. These sensor nodes sense and transmit data to the edge layer for processing, analysis, and decision generation.

***Edge Server Layer*:** This layer is a newly introduced layer into the Safe-aaS platform to enable the distributed computing environment. We consider that the distributed edge server layer and decision layer together form the *distributed edge server end*. Initially, the raw sensed data is transmitted to the edge nodes. After primary processing, these data are processed at the edge servers. Fig. 5.2 demonstrates the mapping of the edge nodes with the edge servers. The mapping of edge nodes to the processed data and the data to the edge servers is a many-to-many relationship. We apply Artificial Neural Network (ANN) to train the processed data for the selection of the appropriate edge server. Further, the decision is generated and propagated to the decision layer using fuzzification. The detailed architecture of the Distributed Edge Server-enabled Safe-aaS platform is illustrated in Fig. 5.1. We categorize this layer into two different sub-layers, which are -

- *Edge Data Computation Layer*: In this layer, different edge devices generate real-time data which are collected, stored, and pre-processed. The pre-processed data are propagated to the next layer for further computation. The edge nodes form the key elements of this layer.
- *Edge Data Intelligence*: The edge servers and decision layer are part of the edge data intelligence layer. Based on the volume of data present at the edge nodes, the edge servers are activated and the decisions are generated. The selection of the appropriate edge server takes place using the ANN process.

***Decision Virtualization Layer*:** Mapping between the generated decisions and the selected decision parameters by the end-users is involved in this layer [19]. Typically, we consider that the processing of this layer is performed centrally in the cloud. Further, the decisions, the present value of decision parameters, and information transferred by the end-users are stored for a long term in this layer.

*Application Layer*: This layer acts as an interface between the Safe-aaS platform and the end-users. The end-users register to the platform, provide their starting and destination details, select the decision parameters, and make payment.

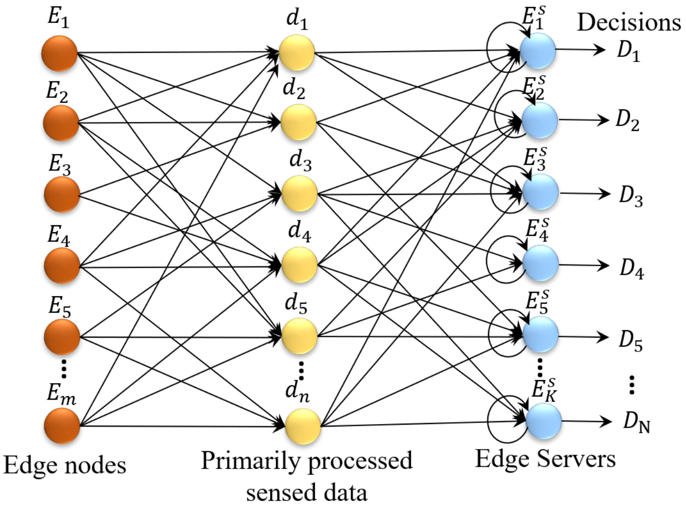


FIGURE 5.2: Mapping of Edge-nodes with The Edge Servers

## 5.3 Characteristics of Edge Server Computing in Safe-aaS

The edge server-enabled Safe-aaS platform provides customized safety-related decisions to the end-users incurring ultra-low latency. We introduce the presence of distributed edge servers in the edge layer of the Safe-aaS platform. The primarily processed sensor data are transmitted to the selected edge server, where the decision is generated. Depending upon the distance between the edge nodes and the edge server, task execution capability, and their storage capacity, the appropriate edge server is selected. Individual computing density is calculated with these parameters at server end. The edge server with positive and high computing density is activated. The chosen edge server is activated to process the data for decision generation. We apply Artificial Neural Network (ANN) to train the sensed data and select the appropriate active edge server. Fig. 5.2 demonstrates the mapping of the edge nodes with the edge server. We represent the computation density of each of these edge servers in terms of their computation capability. The computation capability of the $i^{th}$ edge server is mathematically represented as,

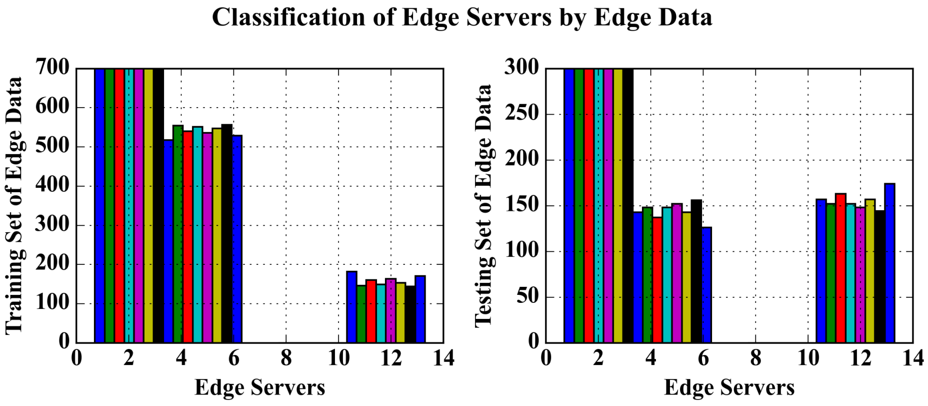
$$C_f^i = \frac{\mathcal{S}_c \times (\mathbb{P}_{max} - (\mathbb{P}_W + \mathbb{P}_r))}{\mathcal{S}_{max} \times \mathbb{P}_{max}} \tag{5.1}$$

where $\mathcal{S}_c$ is the present processor speed and $\mathbb{P}_{max}$ denotes the maximum number of processes to be executed. The number of processes at the waiting state and the number of processes presently being executed at the $i^{th}$ edge server is represented as $\mathbb{P}_w$ and $\mathbb{P}_r$. $\mathcal{S}_{max}$ denotes the maximum processor speed. We model the computing density, $\mathcal{X}^i$ of the $i^{th}$ edge server as a random variable that is normally distributed. Therefore,

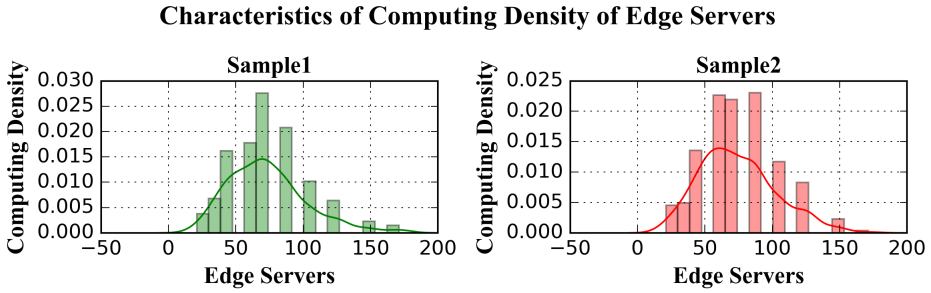
$$\mathcal{X}^i = \frac{C_f^i \times \mathbb{P}_{in}^{i,\tau} \times M_i^l \times t_i}{\mathcal{T}_i} \tag{5.2}$$

where $\mathbb{P}_{in}^{i,\tau}$ represents the volume of data or the number of processes input to the edge server at time instant $\tau$. $M_i^l$ denotes the memory space of the $i^{th}$ edge server. Further, the time taken to process and analyze the data is represented as $t_i$ and $\mathcal{T}_i$ is the maximum time required by the $i^{th}$ edge server. The probability density function of the computing density of the $i^{th}$ edge server is represented as,

$$f(\mathcal{X}_i) = \frac{e^{-\mathcal{X}_i^2/2}}{2\pi} \tag{5.3}$$



(a) Classification of Edge Servers



(b) Computing Power of Edge Servers

FIGURE 5.3: Computation Capability of Selected Edge Servers

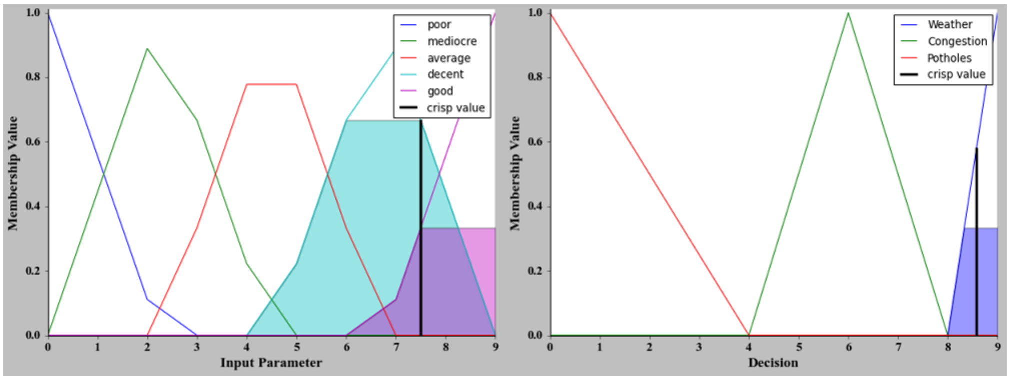


FIGURE 5.4: Fuzzification

$$\mathcal{E}_s^i = \begin{cases} Activated, & \text{if } C_f^i \text{ is high with positive value} \\ & \text{and } \mathbb{P}_{max} > (\mathbb{P}_w + \mathbb{P}_r) \\ NotActivated, & \text{otherwise} \end{cases} \tag{5.4}$$

***Classification of edge servers*:** The edge servers are categorized based on their geographical location and the volume of data processed by them. We identify each edge server and edge node with its unique ID and geographical location. So one edge server $\mathcal{E}_s^i$ is identified by $< \mathcal{E}_{sID}^i, l_s^i >$ and one edge node $e_i$ is identified by $< e_{ID}^i, l_i >$. The Euclidean distance between the edge servers and edge nodes is computed from the geographical location of the edge servers and nodes. The edge servers situated at a minimum distance are selected for processing the data. Further, we apply a supervised Artificial Neural Network (ANN) to classify the edge server through training of available edge data. We consider the scenario where a single edge server is chosen at a particular time to process the data. Thereafter, the excitatory weights of the corresponding links are updated. Fig. 5.3a demonstrates the classification of edge servers with the help of available sensed data at the edge nodes. We take random data as training and testing samples for the classification. After applying the training sample Some of the edge servers are selected. a similar type of result is shown for the testing set. Servers 6 to 10 are not selected because they are not activating with these random data samples. For the testing set The computation capability of these edge servers is estimated which is one of the important criteria in their selection. We compute the computation density of each edge server using Equation 5.2. The edge nodes offload their task to their nearest connected edge server.

***Selection of Appropriate Edge Server*:** To process the end user's request and generate decisions for them, the selection of the appropriate edge server is necessary. Further, the edge servers differ from each other depending upon the distance of these servers from edge nodes and computing density. Algorithm 4 provides a comprehensive overview of the selection of appropriate edge servers. Edge servers are activated through the equation 5.4. We apply the ANN network to train the edge server using the available sensed data at the edge nodes, as demonstrated in Steps

5 and 6. Thereafter, the decision is generated from the primarily processed data transmitted to the selected edge server.

**Algorithm 4** Edge Server Selection

```
INPUT:
 1: Source, Destination, Set of Edge nodes, Set of Edge data, Set of Edge servers.
OUTPUT: Selection of edge server
PROCEDURE:
 1: for Each edge node do
 2:     Calculate Euclidean distance with the edge servers from edge node
 3:     Select edge servers placed at minimum distance
 4:     for Data at edge nodes and selected edge servers do
 5:         Update the link weights of the edge servers
 6:         Train the network
 7:         Appropriate edge server is selected
 8:         Compute computing density of the edge server
 9:         if Computing density is low then
10:             Repeat Step 2
11:         end if
12:     end for
13: end for
```

***Decision Generation at Edge Servers*:** Fuzzification and fuzzy inference is another type of artificial intelligence tool that is used to fuzzify a crisp set of parameters to fulfill the criteria of the decisions. Decision parameters are categorized into five categories based on membership value range. Here we consider $D_8$ to $D_{10}$ as weather type decisions, in the same way, the rest of the decisions are classified as congestion ($D_4$ to $D_8$), another type ($D_0$ to $D_4$). For any specific input value of the parameter, decision $D_i$ is generated. Fuzzified $D_i$ consists of two things- decision type and membership value of the specific type. So the fuzzified set is represented as $\{D_i, \mu_i^d\}$ where $\{0 \geq \mu_i^d \leq 1\}$. Fuzzification is represented mathematically-

$$D_i = \begin{cases} weathertype, & if D_8 \geq \mu_i^d \leq D_{10} \\ Congestiontype, & if D_4 \geq \mu_i^d < D_8 \\ Othertype, & if D_0 \geq \mu_i^d < D_4 \end{cases} \tag{5.5}$$

***Computation of Delay Incurred*:** Without an edge intelligence layer, sensed data is locally processed and then uploaded to the cloud server for further decision generation. Decisions are virtualized through the virtualization layer to the end users or decisions are downloaded. For this scenario, total delay($\mathcal{T}_{d1}$) is calculated mathematically as -

$$\mathcal{T}_{d1} = \Sigma_{i=1}^{m}(\mathcal{S}_d^i \times \mathcal{N}_s^i + \mathcal{P}_d^i + \mathcal{U}_d^i + \mathcal{D}_d^i) \tag{5.6}$$

Total delay is generated for several factors such as delay in sensing the current environment ($\mathcal{S}_d$), processing delay ($\mathcal{P}_d$), uploading delay ($\mathcal{U}_d$), $\mathcal{D}_d$. Here $m$ signifies as number of times the server has been accessed. Delay in processing occurs due to lower processing speed, which is expressed as

$$\mathcal{P}_d^i = \frac{(S_o^i - S_c^i) \times \mathcal{N}_s^i}{t} \tag{5.7}$$

After introducing the edge intelligence layer total delay may occur due to the computation density of the edge server instead of uploading, downloading delay and termed as

$$\mathcal{T}_{d2} = \Sigma_{j=1}^{n}(\mathcal{S}_d^j \times \mathcal{N}_s^j + \mathcal{P}_d^j + \frac{C_f^j \times P}{t}) \tag{5.8}$$

where $P$ is the total number of processes at any time. The total delay for every user is incurred by the number of times the user has to access the edge or cloud server. Total latency is reduced by 90.58% using edge server than using cloud server in Fig. 7.5. Data upload/download to/from the cloud server is not required in our approach. Most of the computations are done at the edge server side.

## 5.4 Performance Analysis

### 5.4.1 Results

In our proposed scheme, edge servers are classified by the edge data with computation capability. The nearest edge server from the edge node is selected for further processing of edge data. Computation density is calculated to mark the edge server as an active server and then perform fuzzification of decision parameters to generate decisions. We characterize the results in the following way:

*Classification of Edge Servers*: Sensed edge data is divided into two sets such as training set and testing set. Here we take 70% of data as a training sample and 30% as the testing sample. The proposed ANN is trained using a training sample with random weights and randomly generated data. In the training process, we get 99% accuracy as shown in Fig. 5.3a and in the testing phase, above 88% of accuracy.

*Computational Power of Edge Server*: It is necessary to determine the computation capability of the selected edge server. Computation capability depends on some parameters such as computation speed, the total number of processes running at this moment, and the maximum number of processes an edge server can process at a time. In Fig. 5.3b, we compare the results for two different samples. We find that computation density is normally distributed and it goes down when the number of edge servers is increased. The possible reason behind this is, that as edge servers are increased, computation is more distributed into several edge servers, so

computing density is decreased.
*Fuzzification*: We categorize the decisions into three categories. Further, we consider 10 decision parameters and 5 generated decisions. Fig. 5.4 illustrates that the fuzzified parameter value 7.5 belongs to the descent and good region. The decision for this input falls under the weather type decision. We apply three rules to make a fuzzified decision.
*Delay Incurred*: We calculate the total delay for our proposed approach. Introducing edge intelligence layer in Safe-aaS, the delay is minimized up to 90.58% than traditional Safe-aaS. Fig. 7.5, the blue line is marked as the delay incurred for traditional Safe-aaS and the green line for Safe-aaS with edge intelligence layer. We observe the increasing trend of delay in both cases in an increasing number of sensor nodes. The possible reason for this trend is as number of sensor nodes usage rises, more decisions are generated. As a result, the more total delay is incurred.

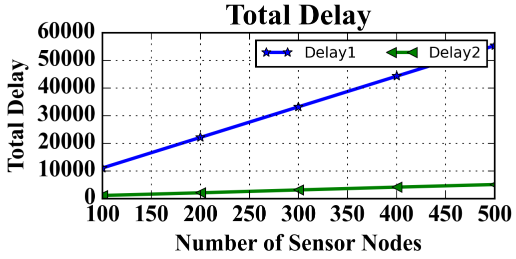


FIGURE 5.5: Delay Calculation

## 5.5 Conclusion

In this chapter, we focused on minimizing decision-generation delays within the Safe-as-a-Service (Safe-aaS) platform, ensuring that end-users receive timely and accurate safety information. Recognizing that delays in generating and delivering safety decisions could compromise the effectiveness of on-road guidance, we designed our platform with mechanisms that emphasize both speed and reliability in information processing and dissemination.

- **Bandwidth Optimization for Data Processing and Storage:** One of the primary challenges in delivering real-time safety information is the need for sufficient bandwidth during data processing, decision storage, and analytics at the network edge. To address this, we proposed an enhanced bandwidth allocation strategy, specifically aimed at expediting the upload of processed data and ensuring seamless storage and retrieval of safety decisions. By focusing on improved bandwidth at critical stages, we aim to reduce data transmission delays and optimize the responsiveness of the Safe-aaS platform.

- **Integration of Distributed Edge Servers:** To further minimize delays and enhance computational efficiency, we introduced the concept of distributed edge servers within the edge layer of the Safe-aaS platform. These servers are strategically positioned to process and store data closer to end-users, thus reducing latency and improving data flow across the network. For effective resource management, we estimated the computational density of each edge server, enabling the platform to assign workloads based on each server's capacity and availability.
- **Application of ANN for Optimal Server Selection:** To facilitate rapid decision generation, we deployed an ANN model at the edge nodes. This ANN model dynamically evaluates the computational demands of incoming tasks and selects the most appropriate edge server based on its processing capacity and current load. By employing an ANN, the system can make intelligent, data-driven decisions in real-time, selecting the optimal edge server for decision generation and thereby reducing processing delays.
- **Fuzzification for Decision Generation at Edge Servers:** To further enhance decision accuracy, we implemented a fuzzification process at the edge servers. This process classifies incoming data based on predefined risk factors and decision parameters, allowing the edge servers to generate safety decisions with greater precision. The fuzzified data facilitates a more nuanced understanding of safety risks, enabling edge servers to produce decisions that are both accurate and timely. Additionally, we analyzed the probability density function of the computational density across the edge servers, offering insights into server utilization patterns and potential areas for improvement.

Further, the proposed edge server-enabled Safe-aaS platform leverages distributed edge computing and ANN-driven server selection to create a responsive and reliable safety recommendation system. By strategically managing bandwidth, computational resources, and decision accuracy, this approach enables the Safe-aaS platform to deliver high-quality, low-latency safety information to end-users, supporting safer and more informed journeys.

# 6
# Adaptive Pricing Mechanism

This chapter introduces a novel smart pricing scheme, 'Safe-Price', designed to enhance Safe-aaS in ITS. The scheme focuses on dynamically adjusting the price charged to users based on the safety-related services they utilize during road trips. Users input their starting and terminal coordinates into a Web portal to request safety information, such as hazardous road conditions, weather updates, and sharp turns. The SSP suggests active decision variables to users based on real-time data from sensors, which they can choose to accept or ignore.

To minimize computational delays in generating safety decisions, an ANN is employed to process sensor data efficiently. The pricing mechanism takes into account the number of decision variables chosen, how frequently users access the service, and whether they follow the SSP's recommendations. Two cases are explored: when users select the suggested safety variables and when they do not. This chapter also highlights the architecture of the Safe-aaS platform and its five-layer SOA, with a particular focus on the role of the decision layer and the application of ANN to improve decision-making speed. ITS have introduced advanced technologies and improved real-time road safety of vehicles and drivers. Abuelela *et al.* [119] proposed that with the growth of vehicular networks, embedded devices, and cloud computing, the concept of vehicular cloud computing, VC2, emerges. On the other hand, proper coordination among the automated vehicles is essential to minimize congestion. The uncoordinated lane change and departure from the location of automated vehicles affect the traffic flow. Taking this fact in consideration, Meissner *et al.* [120] proposed an approach which is based on the organization of the automated vehicles such that the exit success rate of vehicles is maximized with minimum impact on the present traffic conditions. The authors considered different decisive aspects for safe departures such as identification of the vehicles which intend to change their lanes towards exit and identifying the destinations by using vehicle to infrastructure (V2I) service. They proposed the flow mechanisms to maintain balance in traffic flow and safe speed for fuel efficiency. Further, the estimation of right friction on road helps to avoid crash. Also, the autonomous vehicles are not properly designed with anti-skid control. Yuchuan *et al.* [121] collected real-time data and analyzed them by applying deep neural network with domain knowledge concentrating on the texture of road identification. Cloud services are usually applied for processing and management of huge real-time data. End users receive customized safety-related decisions and pay according to their usage in Safe-aaS platform. In Safe-aaS enabled mechanisms [83, 100, 122, 123], vehicles are equipped with wireless sensor nodes to communicate with other vehicles and on-road infrastructures. Considering the different types of road, Roy *et al.* [90] proposed a region-based pricing mechanism for provisioning safety services, such that it satisfies both the service providers as well the users. Further, Pradhan *et al.* [122] categorized the decision parameters as high and low price parameters, to balance the quality of service and price charged from the users using the Safe-aaS platform. In another aspect, Mansouri *et al.* [124] proposed a pricing mechanism through two-level optimization. The authors proposed an optimal pricing strategy for the users through the profit maximization of the cloud service provider. Roy *et al.* [123] formulated an optimization function applying multiple leaders multiple followers Stackelberg game to satisfy both the SSPs and the users. Several research works, related to safety in road transportation, proposed different prototypes, schemes, and platforms [125]. Safe-aaS is one of the prevailing platforms, which delivers personalized dynamic safety-associated decisions dynamically to multiple users. The registered users select certain decision variables and pay the amount charged through a Web portal. During their journey, the users may change the decision variables chosen by them. With the dynamic variation in their selection, the price charged from them also changes. Further, the generation of decisions from the sensed data is a complex task and any kind of delay caused in transmission of information may create an abnormal condition. Therefore, jointly minimizing the delay and delivery of appropriate decisions is a non-trivial problem. Motivated by this fact, we design a pricing scheme for the users depending upon their utilization of safety services. We adopt ANN to process accurate decisions which are delivered to the users. Therefore, safety-based information are simultaneously generated and delivered to the users incurring minimum latency and price charged.

In this chapter, we discuss a pricing scheme to determine the optimal payment charged from the users, as per their service usage. We aim to resolve the following research questions:

- **RQ1**: How to generate safety-related decisions with minimal computation to reduce the latency in the entire process?

- **RQ2**: How do the users access these services?
- **RQ3**: How to determine the optimized price charged to the users?

The most significant contributions of this proposed scheme are as follows:

- We introduce a scheme to jointly minimize the latency incurred in decision generation and the payment received from the users. To increase the usage of the safety services, SSP suggests active decision variables to the users. These parameters are produced from the corresponding activated sensor nodes. On the other hand, the registered users chose these decision variables to receive safety-associated decisions.
- To minimize the processing time, we apply a supervised ANN training-learning technique to generate a decision. We first train the network with the training data set, which consists of active decision variables and generated decisions. Further, based on the selected decision variables, the network is tested and it proves above 80% accuracy between the actual and predicted values in testing samples. We design the utility per user as per the decision variables chosen by them, energy consumed, and time required in decision generation.
- We design a pricing mechanism considering the cases when the registered users select or do not select the active decision variables suggested by an SSP. From their source to destination, the users may modify their selection of decision variables. Based on the selected variables, the price is received by them. In our proposed approach, the payment received is decreased from 58% to 36%.
- Extensive mathematical and simulation-based evaluation of the proposed scheme explores that the sensor usage in decision generation reduces. Therefore, the energy consumed in the generation of a decision is minimized and the service charge received from users also fluctuates with their selection of decision variables.

## 6.1 Problem Definition

We consider an ITS-implemented environment, where Safe-aaS platform is applied. Safe-aaS is a five-layered platform - device, edge, decision, decision virtualization, and application layers. The device layer contains heterogeneous types of stationary and mobile sensor nodes, which sense and transfer the data to the edge layer/cloud, depending upon the time-sensitive nature of the data. Practically, the sensor nodes are either placed at different geographical positions or into the vehicles, which are activated at various time intervals. Based on their energy levels and reputation, the activated sensor nodes are selected for decision generation. Further, the decision is generated from the sensed data transmitted by various sensors. Therefore, depending on the type of activated sensor nodes, the corresponding decision variables are determined and suggested. The registered users, choose some the available decision variables and pay the amount charged for availing the safety services through a web portal. Based on their preferred decision variables, customized decisions are provided to them. On the other hand, a decision is generated after complex analysis of data in the decision layer and virtualized to users. Then, we apply ANN in this layer to train the pre-processed data generated from the edge layer/cloud. The logical mapping between the generated decisions and decision variables requested by the users is performed in this layer applying ANN approach.

We consider $\mathcal{N}$ as the set of different types of sensors in the device layer, which is represented as $\mathcal{N}$=$\{\mathcal{S}^1, \mathcal{S}^2, \cdots, \mathcal{S}^N\}$, such that $1 \leq i \leq N$. Further, we categorize the sensor nodes based on the type of data sensed by them such as weather conditions, traffic congestion, and presence of pedestrians. We consider that $x$ types of sensors working in the environment. Therefore, the number of activated sensor nodes is mathematically represented as,

$$N_s = \{\mathcal{S}_i : \mathcal{S}_i \rightarrow (S \vee M), \mathcal{S}_i \in C_x, \mathcal{S}_i \in \mathcal{N}\} \tag{6.1}$$

where $S$ and $M$ represent the stationary and mobile sensor nodes. The category or type of sensor node is denoted by $C$. We define the *reputation* of the sensor nodes in terms of the number of times these nodes are transmitting the sensed data with respect to the normal sensing data rate. Therefore,

$$\mathcal{R}_i = \frac{n_i}{n_{th}} \times r_s \tag{6.2}$$

where $\mathcal{R}_i$ is individual reputation of the $i^{th}$ sensor, $n_i$ is the frequency of transmitting sensed data, $n_{th}$ is the threshold frequency to transmit sensed data, and $r_s$ is the rate of sensing. As mentioned earlier, practically, at any time instant, some of the sensor nodes are activated. Mathematically,

$$\mathcal{S}_i = \begin{cases} \text{Active}, & \text{Energy level/Reputation value is high} \\ \text{Inactive}, & \text{Energy level is low and needs power} \end{cases} \tag{6.3}$$

Sensors are activated when the energy and reputation of the sensor nodes are above their threshold value. The corresponding available decision variables from these activated sensor nodes are denoted as $P_a^{ij}$. Each of these activated decision variables belong to the set of available decision variables $\mathcal{P}$, such that $P_a^{ij} \in \mathcal{P}$. In case the $i^{th}$ decision parameter, $P_a^{i,j}$ is active, a random weight is generated for it. The weightage of each decision variable belongs to the range 0-1. Further, depending upon learning rules, these values are updated. During the training of decision variables, the weight update takes place, till the decision is generated. We

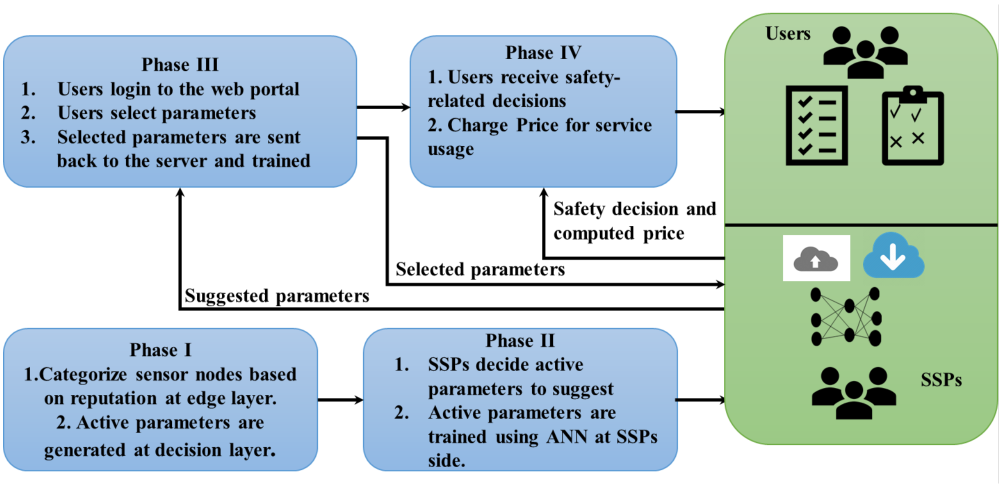


Figure 6.1: Safe-Price: Block Diagrammatic Representation

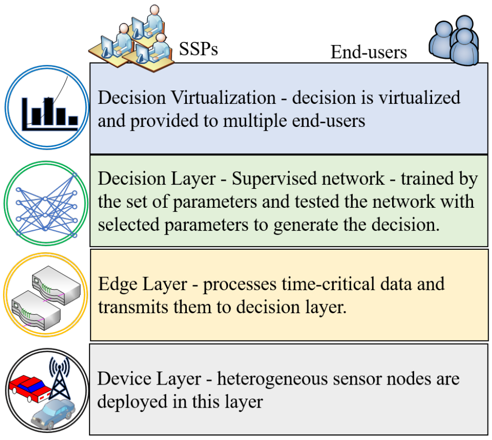


Figure 6.2: Safe-Price: The Layers and Their Functions

represent the set of users as, $\mathcal{E} = \{E^1, E^2, \cdots, E^m\}$. Based on the number and type of decision variables chosen by users, the decision is generated. The value of these decision variables are also updated, after a duration of $T_p$. On the other hand, ANN is applied for decision generation in the decision layer, the entire process is discussed in Section 6.1.1.

### 6.1.1 Decision Generation

In our proposed pricing mechanism, we apply supervised ANN on the primarily processed data. The input layer comprises the available active decision parameters. Thereafter, we define two hidden layers and set the weights with random values. For the activated decision variables, the weights are updated immediately. First, we train the primarily processed data generated by the heterogeneous sensor nodes. Then, the weights of their corresponding decision variables are updated. These decisions are further classified, depending upon the category of sensors. We consider that certain sensor nodes are activated within a time duration. Therefore, the corresponding decision variables also become active during that time. The training is undergone with these active decision variables for our randomly generated sample data set. Depending upon the decision variables selected by the users, the decision is generated after training. We represent the number of decisions to be generated as a set, $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_x\}$.

*Training Process:* ANN is trained with the active decision variables, corresponding to the sensed data, at the input layer, and the decision is generated at the output layer. Considering the energy level of the sensor nodes, the weights of the decision variables are set by the SSP. Suppose, the hidden layer comprises $m$ neurons. The weight vectors of the hidden and output layer are denoted as $W = \{W_{11}, W_{12}, \cdots, W_{nm}\}$ and $Z = \{Z_{11}, Z_{12}, \cdots, Z_{mx}\}$ respectively. Every hidden neuron uses the sigmoid function on the weighted sum from input layer. The stream of output vectors from the input layer which are propagated to the hidden layer is denoted as $\{Y_{11}, Y_{12}, \cdots, Y_{nm}\}$, and $Y_{ij} = \sum_{i=1}^{k} P_{ij}^{a} W_{ij} + b_j$, where $k$ is the number of decision variables chosen by the $a^{th}$ user. Suppose, $H_j$ is the set of outputs generated from the hidden layer. After applying sigmoid function, the generated decision at output layer is $V_j = \sum_{j=1}^{q} H_j Z_{jx} + b_x$, where $q$ is the number of outputs combined to generate the decision. After applying the linear activation function at the output layer, the network provides the final output. Further, we calculate the mean square error as,

$$\mathbb{E}_j = \frac{1}{2} \times (f(V_j) - D_j)^2 \tag{6.4}$$

Motivated by the mathematical model of ANN in [126], parameters are updated in terms of weights and bias by the learning process.

We consider that before occurring any hazardous situation, specific sensor nodes are activated and decision variables are generated from the sensed data. These data are analyzed, the generated decision is transmitted to the registered users. In our proposed approach, we compute the usage of the safety services, based on number of times and time duration, the users access the service. The utility of the $j^{th}$ end-user is -

$$\mathcal{U}_j = \sum_{i=1}^{N} f(P_a^{ij}, \varepsilon_i^{eff}, \mathbb{T}_i^{eff}) \tag{6.5}$$

where $P_a^{ij}$ is the decision variables selected by the $j^{th}$ end-user. We consider that any end-user selects a maximum of $N$ decision variables. $\mathbb{T}_i^{eff}$ represents the time required to process the users' requested decision parameters. Therefore,

$$\mathbb{T}_i^{eff} = \frac{(\mathcal{T}_p^i + \mathcal{T}_t^i + \mathcal{T}_r^i - \mathcal{T}_{ia} - \mathcal{T}_d^i)}{T_i} \tag{6.6}$$

**Algorithm 5** Price and Utility Computation

```
INPUT: ⟨Starting, Destination⟩ = $\langle S_i, d_i \rangle$, $P_aij$, $P_s$.
OUTPUT: Service charge payment by $i^{th}$ user.
PROCEDURE:
 1: for $n$: Number of users do
 2:     for $K$: Number of active decision variables do
 3:         $i^{th}$ user selects decision variables
 4:         $P_s^i$ is computed
 5:         Call Algorithm 6 for decision generation
 6:         Per selection price charged and utility is computed
 7:         while time = $t_p$ do                          ▷ $t_p$: short time delay
 8:             Go to step 2 until $d_i$ is reached.
 9:             if $i^{th}$ user satisfied then
10:                 Decision is delivered
11:                 Total Price is estimated
12:             else
13:                 Jump to line no. 5
14:             end if
15:         end while
16:     end for
17: end for
```

where $\mathcal{T}_p$, $\mathcal{T}_t$, and $\mathcal{T}_r$ denote the time period for which the selected decision parameters are active, time required to transmit the sensed value of the decision variables, and the response time for each decision parameter, respectively. The $i^{th}$ decision parameter becomes inactive beyond the time duration $\mathcal{T}_{ia}$. Additionally, we consider that a delay of $\mathcal{T}_d^i$ is incurred to process the decision parameter. The maximum time required to process the decision is denoted as $T_i$.

The decision variables are active for a time duration $t_l^i$, beyond which the energy level of these sensor nodes tends to decrease. We consider that $t_l^i \leq t_{eh}^i$ where $t_{eh}^i$ is the time duration for which the energy level of the $i^{th}$ sensor node is high. $\mathbb{E}_i^{eff}$ is denoted as the effective energy of the sensor nodes corresponding to the decision variables. Mathematically,

$$\mathbb{E}_i^{eff} = \frac{\mathbb{E}_i^{resi}(N_{\mathcal{S}_a} - N_{\mathcal{S}_s}) + (\mathbb{E}_i^{sens} + \mathbb{E}_i^{trans})}{\mathbb{E}_i^{init} \times N_{\mathcal{S}_a}} \tag{6.7}$$

where $\mathbb{E}_i^{resi}$, $\mathbb{E}_i^{sens}$, $\mathbb{E}_i^{trans}$, and $\mathbb{E}_i^{init}$ represent the residual energy, energy consumed for sensing, energy required for transmission, and the initial energy of the $i^{th}$ sensor node. The number of active decision variables and chosen decision variables by the users are represented as $N_{\mathcal{S}_a}$ and $N_{\mathcal{S}_s}$. In this proposed approach, we primarily focus on satisfying both the SSPs and the users, such that both are benefited. The SSPs provide the generated decisions to the users, and claim the payment from them, during which the service is requested.

### 6.1.2 Pricing Mechanism

In this proposed scheme, we consider that the SSP recommends some active decision variables to the registered users, depending on their requirement. The sensor nodes possess limited energy, therefore, the decision variables corresponding to them become inactive after a certain time period. Moreover, the sensing, transmission, processing, and decision generation take place in real time. We assume that the processing charge, $C_p$, is quite high for these active decision variables. Further, the users are charged, depending upon the number of times they availed the safety services. We consider three cases of pricing, which are as follows:

*Case I*: Suppose the end-user initiates his/her journey at time instant $t_1$. The SSP suggests active parameters, among the ones available in the Web portal. In case, the end-user does not select any active decision variable advised by the SSP, then minimum service price is charged. Since the Safe-aaS platform does not process the active parameters in this case, processing charges are not included with the price charged. Therefore,

$$C^j_{tot} = \sum_{i=1}^{(M-K)} (\mathcal{N}^i_a C_s + ((\mathcal{P}_a - \mathcal{P}^{ij}_s)C_p) - C_d) + (KC_{min}) \tag{6.8}$$

where $C_s$ is the price charged for the active sensors used $\mathcal{N}^j_a$. $\mathcal{P}^j_s$ is the number of decision variables chosen by the $j^{th}$ end-user. $C_p$ represents the processing charge incurred to generate the decision. We assume that a discount, $C_d$ is allowed by the SSP for selecting safety services. The SSP recommends active parameters $M$ times and the end-user did not select any of the parameters $K$ times. Therefore, minimum charges $C_{min}$ is incurred, which depends on $K$.

*Case II*: When the end-user agrees with the decision variables recommended by the SSP, the total payment received is represented as,

$$C^j_{tot} = \sum_{i=1}^{M} \mathcal{N}^i_a \times C_s + (\mathcal{P}_a - \mathcal{P}^{ij}_s) \times C_p - C_d \tag{6.9}$$

*Case III*: In a dynamic scenario, the users may have the option to select/remove any previously chosen decision variables at different points from their source to destination. This is a hybrid form of previous two mechanisms. In case the users select the decision variables, then pricing is calculated as per Case I. On the other hand, if the users remove any selected decision variables, then pricing is calculated as per Case II. But total price charged by SSP includes all the prices calculated during journey.

### 6.1.3 Profit Analysis

In our “Safe-Price" scheme, the SSP acts as the centralized entity and is responsible for controlling the end-to-end safety services. Profit of SSP is determined by estimating the cash outflow and inflow on the basis of the number of users, sensor types, number of active sensors and their geographical location. We consider **L** as the set of geographical locations which comprises $l$ locations within the same SSP region. If **K** categories of sensor nodes are working on decision generation, then the number of sensor nodes involved in each category is considered as variable $\mathbf{N}^i$, where $1 \leq i \leq \mathbf{k}$. The location-wise cash outflow is mathematically represented as,

$$C_{out}^{ssp} = \sum_{l,k}^{i=1,j=1} \mathbf{L}_i \times \frac{N_s^i}{\sum_m^{c=1} \mathbf{N}_c} \times C_s \tag{6.10}$$

Further, the cash inflow is the price charged from the end-user at any particular location. Therefore, the profit, $\mathcal{P}$ of SSP is computed as-

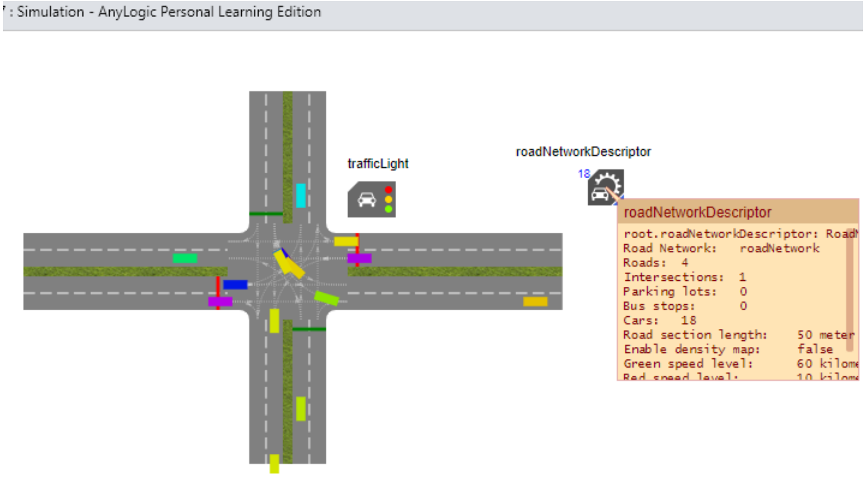
$$\mathcal{P}^{ssp} = \sum_{n,l}^{i=1,j=1} \left( (\mathbf{L}_i^j \times C_{tot}{}_i^j) - C_{out}^{ssp} \right) \tag{6.11}$$

Algorithms 5 and 6 depict the overview of payment received from the users and decision generation procedure. In Step 3 of Algorithm 5 the end-user selects the decision variables. Based on their selected decision variables and those suggested by the SSP within time $t_p$, the price charged and their utility is computed, in steps 7 to 12. On the other hand, we apply supervised ANN for the decision generation. In Algorithm 6, the neural network is trained from steps 2 to 4. Further, the network is tested with data sets and error is calculated in steps 6 and 7. Thereafter, the decision variables are updated, depending upon the error in steps 8-11. The generated decision acts as input in step 5 of Algorithm 5.

**Algorithm 6** Decision Generation

```
INPUT: $P_a, P_s$.
OUTPUT: Decision for $i^{th}$ end-user.
PROCEDURE:
 1: Create MLP neural network with hidden layers
 2: for $N$: Number of training sets do
 3:     Train MLP neural Network
 4: end for
 5: for $k$: number of testing sets do
 6:     Test the network with testing sets
 7:     Error estimation
 8:     if error=true then
 9:         Update parameters of the network
10:     end if
11:     Possible decision is generated
12: end for
```

*Time Complexity Analysis of The Algorithm:* In Algorithm 5, we are calculating price



```
root.roadNetworkDescriptor: Road
Road Network:    roadNetwork
Roads:  4
Intersections:  1
Parking lots:   0
Bus stops:      0
Cars:   18
Road section length:    50 meter
Enable density map:     false
Green speed level:      60 kilom
Red speed level:        10 kilom
```

FIGURE 6.3: Vehicular Mobility

charged from the users. The $K$ decision parameters are modified after every $t_p$ delay in total journey until destination reached. This means if first parameter is suggested at $t_p^0$ time, then next suggestion takes place at $t_p^1$ and $t_p^2$ respectively. Therefore, until destination is reached, it takes $log(t_p)$ time for certain iterations. In Algorithm 6, decision is generated for the $P_s$, selected parameters, through the MLP layers - input, hidden, and output. Output layer provides linearity in decisions, so it takes $O(n^2)$ time, in the worst case. The overall time taken for $N$ users is - $\mathbb{T}(N) = N \times l_t \times (n^2 + k + c)$, where $k$ is the time taken for testing dataset in Algorithm 6 and $c$ is the constant time required for calculating price and utility. Therefore, overall time complexity is $O(n^2 \times log(t_p))$ per user. In worst case, overall time complexity is $O(n^2)$.

## 6.2 Performance Evaluation

### 6.2.1 *Simulation Design*

In Safe-Price mechanism, we use Python as programming tool and Anylogic as the emulator tool to simulate the proposed environment. We consider a road segment of 100 meters with one intersection point. It is developed with traffic flow mechanism. Anylogic is multi-method modeling tool. Models are created for a number of industries, including supply chain management, healthcare, and traffic. Fig. 6.3 illustrates the mobility of the vehicles through intersection. We apply traffic signals at the intersection point to control the flow of vehicles. Road network descriptor describes the various traffic-related information such as speed of the cars, number of cars moving on the road. In our simulation environment, we vary the number of users from 50 to 250 in the presence of 1000 randomly deployed stationary and mobile sensor nodes for the performance evaluation. The various parameters are analyzed by extensive simulation and they are listed in Table 6.1.

TABLE 6.1: Simulation Parameters

| **Parameters** | **Values** |
|---|---|
| Active Parameters | 15 |
| Sensor Nodes | 1000 |
| Minimum parameters selected | 4 |
| Cases for price charged | 2 |
| $C_a$ | 100 |
| $C_p$ | 100 |
| $C_d$ | 0 – 100 |
| $C_{min}$ | 200 |

### 6.2.2 Benchmark Solution

We analyze the performance of the proposed pricing mechanism, Safe-Price in two perspectives - profit of SSP and utility of the users. We compare “Safe-Price" with three existing pricing schemes - Per-Subscriber model [97], Reg-Price [90], and PRIME [108]. Guijarro *et al.* [97] proposed a both-sided pricing scheme, which acts as the link between the users and Wireless Sensor Networks (WSNs). The proposed per-subscriber model considered the cost paid by users and cost paid to WSNs, such that their profit is maximized. In “Safe-Price" scheme, we compute the total utility, considering the two cases - users are not accessing the service and users accessing the service both. We consider another regional-based pricing scheme proposed by Roy *et al.* [90] as the benchmark. The authors calculate the utility in terms of sensing zones, responsiveness and sensor categories. Similarly, we also consider that the vehicles are moving through various geographical regions. However, we calculate the utility of users in terms of the decision variables selected, residual energy of the sensor nodes, and time duration. In another pricing scheme, Roy *et al.* [108] proposed a pricing mechanism for provisioning mobile Sensors-as-a-Service (mSe-aaS).

Fig 6.4 illustrates the fluctuations in the utility of the existing and proposed schemes. It is discovered that the utility has a declining tendency in comparison to the rising trend of consumers. From the five iterations undergone during simulation, we observe that the utility of the users in Safe-Price is improved by 67%, 70% and 50% (approx.) compared to PRIME, Per-Subscriber, and RegPrice respectively. However, fig. 6.5 demonstrates the fluctuations in the profit of the SSP in five iterations. We noticed that the profit of service provider is improved by 4%, 65%, and 15% (approx.) compared to PRIME, Per-Subscriber, and RegPrice respectively. The overall sensor usage to generate a decision, is minimized in Safe-Price. Therefore, the cash outflow is reduced, and the profit of SSP is enhanced from the other existing schemes. Additionally, the energy consumed in decision generation is minimized.

*Performance Analysis of classification Models in Decision Generation:* We evaluate the various machine learning models with our random sample dataset. The dataset contains 15

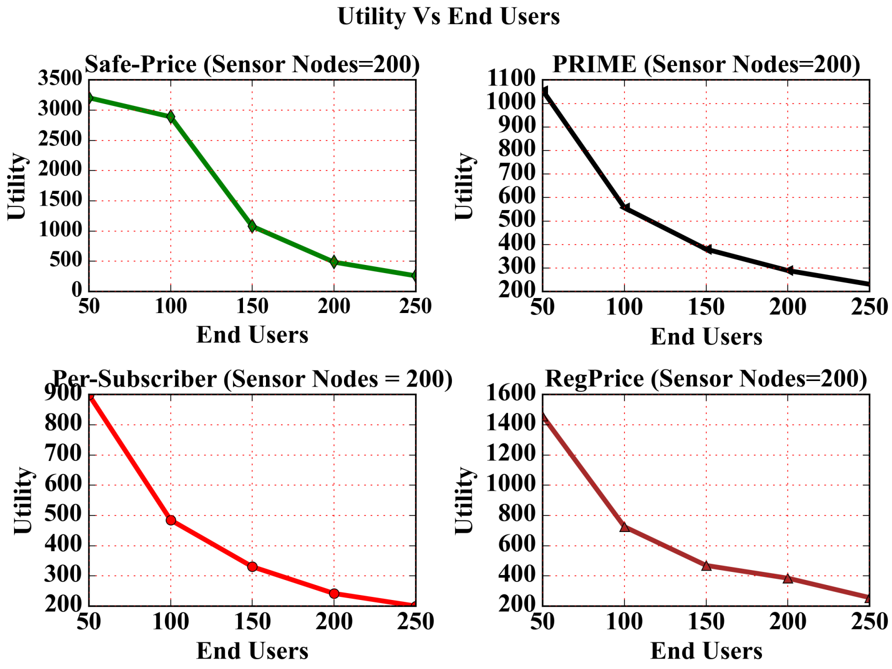


FIGURE 6.4: Utility Per User

decision variables and 5 decisions. We split them into training and testing datasets with 70% and 30% data respectively. We compute the performance score for the classification/regression models to analyze the datasets. Table 6.2 demonstrates that all the models are performing well except Support Vector Machine, for testing dataset. However, all these models other than ANN are not working properly on unknown or validation data. Our randomly generated dataset contain nearly 800 samples. The testing dataset contains 230 samples and the validation dataset contains 30 samples. Though all models are responding for the test samples, but are not performing well for unknown dataset samples.

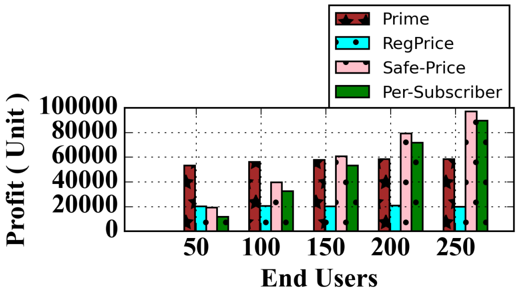
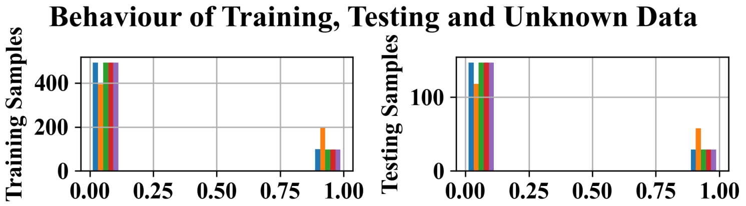


FIGURE 6.5: Profit of SSP

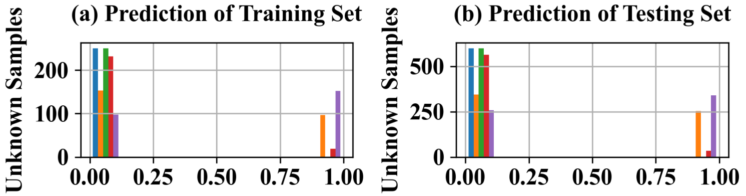
TABLE 6.2: Classification Report of Various Models

| Models | Testing Dataset | | | | | Validation Dataset | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Precision | Recall | F1 | Support | Accuracy | Precision | Recall | F1 | Support | Accuracy |
| ANN | 0.99 | 0.99 | 0.99 | 230 | 1.0 | 0.85 | 0.85 | 0.85 | 30 | 0.99 |
| RF | 1.0 | 1.0 | 1.0 | 230 | 1.0 | 0.03 | 0.028 | 0.028 | 30 | 0.05 |
| KNN | 1.0 | 1.0 | 1.0 | 230 | 1.0 | 0.26 | 0.405 | 0.302 | 30 | 0.5 |
| SVR | 1.0 | 1.0 | 1.0 | 230 | 1.0 | 0.56 | 0.31 | 0.38 | 30 | 0.74 |
| MLPR | 1.0 | 1.0 | 1.0 | 230 | 1.0 | 0.6 | 0.2 | 0.3 | 30 | 0.2 |



(a) Prediction of Training Set

(b) Prediction of Testing Set



(c) Prediction of Unknown Set

(d) Prediction of Unknown Set

FIGURE 6.6: Prediction

### 6.2.3 *Result Analysis*

We evaluate the overall performance in various aspects.

*Decision generation:* In our proposed approach, the decisions are generated through basic training-learning process with randomly created data set. We observe that the training-learning process is successfully performed with accuracy score 1.00 using the training data set, 0.99 in testing and above 0.9 in unknown data set. Fig. 6.6 illustrates how the training-learning is evaluated with the training, testing, and unknown data set. During the process of decision production, we analyze the various parameters used for the evaluation of service utility.

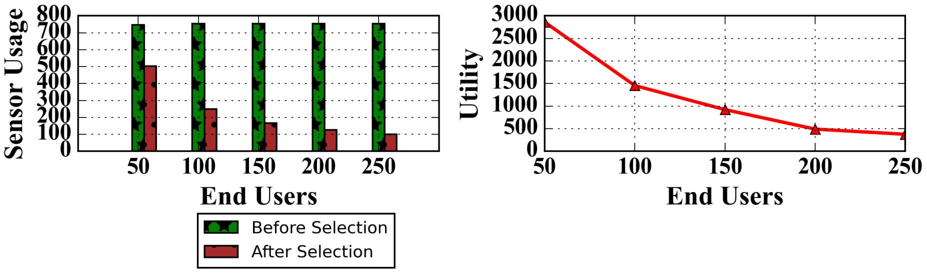


FIGURE 6.7: Sensor Usage and Utility of users

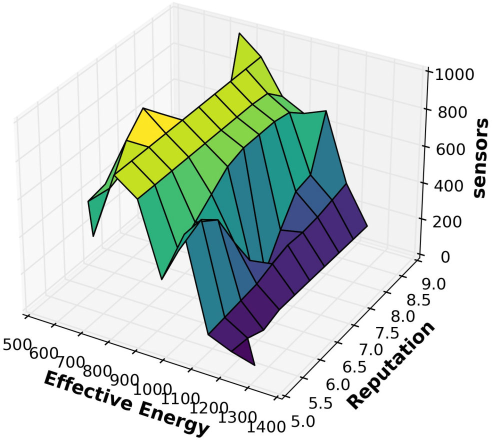


FIGURE 6.8: Activated Sensors

*Sensor Usage and Utility*: In this pricing scheme, our primary aim is to minimize the charging price from the users, the energy utilized in decision generation through the reduction of sensor usage. Further, we minimize the overall utility per user in respect of energy consumed and sensor nodes used. Fig. 6.7 demonstrates the fluctuation in the usage of sensor nodes, before and after the selection of SSP's suggested active decision variables by the users. As recommended by the SSP, the utilization of sensor nodes is reduced from $60-68\%$ to $20-30\%$ in the five iterations in incremented number of users. Additionally, in Fig. 6.7, we observe that after considering the active decision variables recommended by the SSP, the number of acting sensors decreases. The overall consumable energy decreases by almost 80% with the iterations. The possible reason behind such trend is that the only sensor nodes corresponding to the active decision variables are utilized for decision generation. Further, the sensor nodes are activated based on their reputation and effective energy after certain threshold value at that particular moment, as discussed in Fig. 6.8.

*Pricing Analysis:* In “Safe-Price" scheme, the SSPs recommend some significant active decision variables to the users. Further, service charge payment by the users, is directly related to the variable selection at any instant. As a result, sensor usage in the decision generation fluctuates. We consider three cases - one, when the end-user does not avail safety services, when the end-user selects the suggested decision variables and when users consider mixed approach. In Fig. 6.9 we found that the average service charge from the users trend to decrease by 36% with the increased number of users. On the other hand, Fig. 6.9(b) illustrates the variations in the price charged, when the users do not select the SSP's suggested decision variables. We observe that the average price received from them is much less than the previous one throughout their journey. Fig. 6.9(c) demonstrates a comparison in the payment received from users, when the service is availed and not availed by them. We observe that the payment against the service usage, received from the users is decreased from 36% to 58%.

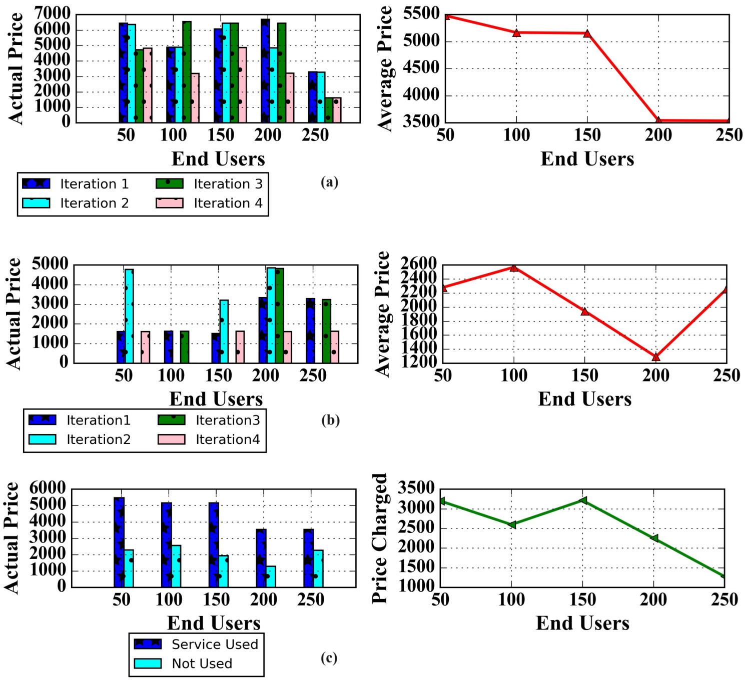


FIGURE 6.9: Pricing-Cases: (a) Case I (b) Case II (c) Case III

## 6.3 Conclusion

In this chapter, we introduced a dynamic pricing scheme aimed at facilitating accurate safety-related decisions for users while optimizing the utilization of sensor nodes. The primary objective of our approach was to strike a balance between reducing the operational usage of sensor nodes and maintaining high accuracy in the decisions generated.

To achieve this, we employed a supervised ANN approach for training and decision-making. The scheme includes an SSP that recommends specific active decision variables to users. Users then select these variables during their journey, from the origin to the destination. Based on the chosen active decision variables, the corresponding service charges are applied to the users. Importantly, only the sensor nodes linked to the selected variables are activated, resulting in significant optimization of sensor usage.

The effectiveness of our proposed scheme was thoroughly validated through a series of extensive simulations that were carefully designed to assess various performance metrics. The results obtained from these simulations indicated that the profit margins associated with the SSP are significantly influenced by the geographic locations in which they operate, as well as by the total number of sensors that are deployed within those areas. Furthermore, our observations revealed a clear trend: as the number of iterations in our testing increases, both the residual energy levels of the sensors and the overall usage rates of the sensors show a consistent decline. This trend underscores the efficiency of our approach in the management and allocation of resources within the network. Ultimately, this dynamic pricing model that we have implemented not only ensures a high degree of cost-effectiveness but also actively promotes the sustainable operation of sensor networks over time, thereby contributing to their long-term viability and efficiency.

In future works, we have plans to introduce a novel decision-generation scheme aimed at significantly reducing the computation delay that can occur within a 6G environment. This innovative approach will take into account the unique challenges and requirements of such advanced network technology. Additionally, we intend to develop a comprehensive scheme that will enable us to thoroughly analyze the overall Quality of Service (QoS) delivered by the SSP throughout the various message communication links. This analysis will be based on a range of critical network parameters, including but not limited to response time delay, packet delivery ratio, and throughput. Each of these parameters plays a vital role in assessing the performance and reliability of the network. Moreover, ensuring safety recommendations that encompass all these factors is an essential aspect in the context of a Safe-as-a-Service (Safe-aaS) environment, as it is crucial for maintaining high standards of service and user trust.

# 7

# Safety Recommender System

This chapter presents a framework for a Safety Recommender System that offers Safe-aaS in IoT-based transportation systems, aiming to enhance real-time safety for on-road vehicles. Timely and accurate delivery of safety-related information to road users can greatly reduce the risk of accidents. Existing research efforts focused on enhancing road safety through different technological and methodological approaches. Real-time safety parameters encompass both environmental factors and vehicle-related safety, including compliance with certain road standards. Pecht *et al.* [127] developed a safety detector system aimed at identifying defects in vehicles, which is critical for ensuring that vehicles meet safety standards. The challenge of identifying pedestrians in the IoV environment has also been a focus of research, given the increasing complexity of urban traffic. Sharma *et al.* [128] proposed a system designed to detect pedestrians approaching vehicles, contributing to the on-road safety of pedestrians, a group that is particularly vulnerable in traffic environments. Driver drowsiness has been identified as another significant contributor to road accidents. Caban *et al.* [129] analyzed the causes of accidents and emphasized that drowsy driving is a critical factor that needs to be addressed to improve road safety. In addition to physical safety measures, the behavior of drivers is a key factor in road safety. Fernandez *et al.* [40] proposed a framework and a fuzzy rule-based system to classify driver behavior, which can help in identifying risky driving patterns. Similarly, Kaplan *et al.* [41] explored the various factors that contribute to driver distraction, another major cause of road accidents. Martinez*et al.* [130] further studied driving behavior and classification, contributing to a better understanding of how driver actions impact road safety.

Extreme weather conditions pose additional challenges to road safety, and Intelligent Transportation Systems (ITS)-based solutions have been proposed to mitigate these effects. Dey *et al.* [131] and Cools*et al.* [132] investigated ITS-based services designed to maximize on-road safety despite adverse weather conditions. Nassar *et al.* [133] also studied the impact of dynamic weather conditions on road safety, emphasizing the need for adaptive safety measures in changing environments.

AI-based techniques have been increasingly applied in this field, particularly in the development of recommender systems. Saranyadevi*et al.* [134] developed a fuzzy-based recommender system that analyzed accident datasets to uncover various factors influencing road safety. Soumya *et al.* [135] provided an Intelligent Traffic Management system combined with a recommendation system, showcasing the potential of AI to enhance road safety through data-driven insights. In this chapter, we discuss a decision recommendation system that takes into account various factors such as weather, road conditions, and the driver's risk profile.

The proposed system integrates several phases, including data collection from sensors, self-supervised learning, and the generation of personalized safety plans. The system analyzes dynamic on-road conditions, weather patterns, and driving assistance parameters and provides tailored safety recommendations to registered users through the Safe-aaS platform.

In an IoT-based transportation system, vast amounts of data are generated by sensor nodes, creating challenges related to data storage, processing, and analysis, especially in edge devices. To address this, the system utilizes a fuzzy rule-based learning approach to optimize safety recommendations, ensuring minimal processing delay and low latency in delivering safety-related decisions. A safety score is computed from the pre-processed data, followed by a user-specific similarity score and risk score to create customized safety plans. The chapter also introduces a user-friendly application interface for real-time implementation of the system.

Extensive analysis demonstrates the system's effectiveness in detecting on-road risks and adjusting recommendations in dynamic environments, such as varying road conditions and weather. By delivering timely safety decisions, the recommender system plays a crucial role in minimizing road accidents and improving overall vehicle safety in IoT-based transportation.

## 7.1 Recommender System for Safe-aaS

Autonomous vehicles are mostly equipped with various types of sensor nodes. These sensor nodes sense, and generate enormous information regarding driver behavior and the surrounding environment. Generated data are analyzed to improve the on-road safety of the drivers. We consider a Safe-aaS implemented scenario, which comprises a cloud platform for the storage and analysis of the generated data. Safe-aaS is a five-layered architecture - device, edge, decision, decision virtualization, and application. The safety-related decisions are generated in the decision layer of Safe-aaS architecture. Further, these decisions are transmitted to the decision virtualization layer, where logical mapping between decisions and decision variables, takes place. On the other hand, in an IoV environment, the information is shared among the registered and neighbored users in real time. Therefore, the latency incurred in the delivery of decisions to a huge number of users is minimized. We propose a framework for a decision recommender system that provides customized safety services. The values of the decision parameters are stored in the cloud server database. Further, the risk values are calculated for the dataset considering different parameters. Fig. 7.1 illustrates the proposed recommender system, which works at the sub-layer of the decision layer in Safe-aaS. We observe that the vehicles in the device layer are interconnected. The registered users receive a safety plan, which they share with neighboring registered users as they are wirelessly connected. The major key processes of the proposed recommender system are: (a) storing and prepossessing of sensed data, (b) essential features extraction by self-learning, (c) possible decision parameters are suggested by SSP to the users and the safety score of the safety measures is computed simultaneously, (d) the registered users select parameters and the risk value is calculated for them, and (e) if SSP approves, then safety plan is provided to the users. (f) Each of the registered users maintains a user profile that contains GPS location, destination, transport registration ID, and past record of any accident. (g) If the risk value is greater than the safety score of the corresponding zone of the location, then a safety plan is recommended with different safety measure values to the users. (h) SSPs validate and optimize the safety plan by minimizing errors between the safety score and risk score. Self-learning is important to learn the system with already known facts or datasets. A fuzzy rule-based system learns from the rules defined and provides fuzzified risk values. Thereafter the generated risk values are compared with the calculated risk score of selected parameters by the users. In Fig. 7.2, we describe the fuzzy learning phase. In this phase, crisp values of different parameters are inserted as the input and then converted into fuzzified values by the triangle fuzzification method. Lastly, the rules are applied to learn and generate the output.

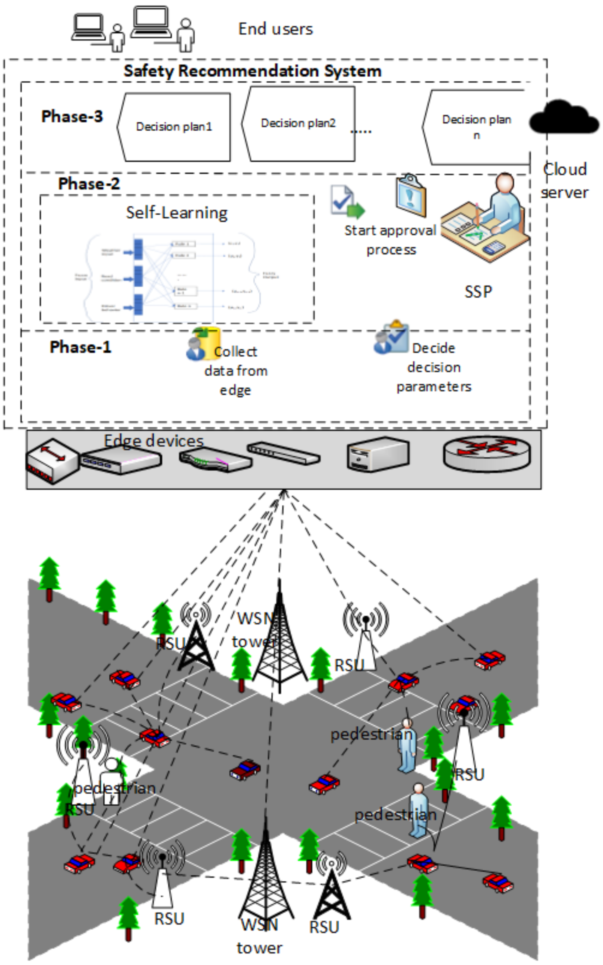


FIGURE 7.1: Proposed Framework

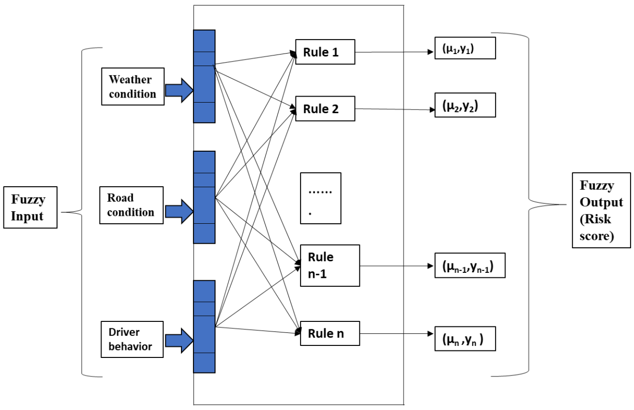


FIGURE 7.2: Self-Learning

## 7.2 Risk Assessment and Safety-Decision Generation

In Safe-aaS, the registered users are recognized with their unique ID, which contains $\langle l, d, regID, pasthistory\rangle$, where $l$ is the current location, and $d$ is the destination location. SSPs are aware of the working zone from the current location. Sometimes the distance between $l$ and $d$ is split into several working zones. *RegID* is the transport registration number along with the safe-aaS registration number whereas past history of the vehicles contains vehicle condition details and driving score. The higher the value of the score indicates a more safe and skilled driver. Suppose the total distance between $l$ and $d$ be divided into $\{z_1, z_2, \cdots, z_k\}$ operating zones of the SSP. Suppose, $\mathcal{U} = \{u_1, u_2, \cdots, u_n\}$ is the set of registered users. We consider different safety measures for each zone, which are stored in the cloud server database. Each sensor data is pre-processed at the edge devices and then transferred to the server. Suppose road condition data set is $\mathcal{R} = \{r_1, r_2, \cdots, r_a\}$, weather condition data set is $\mathcal{W} = \{w_1, w_2, \cdots, w_b\}$, and driver condition data set is $Dr = \{dr_1, dr_2, \cdots, dr_c\}$. The safety score of a zone, $i$, is estimated as,

$$f(z_i) = (SC(\mathcal{R}_i) \times SC(\mathcal{W}_i) \times SC(Dr_i)) \mod Rs_i \tag{7.1}$$

Where $Rs$ is the previous risk value of a particular zone, $SC$ is the safety score considering each of the safety measures. Each of the users selects the decision parameters from all types of safety measures, during registration. We assume a set of decision parameters generated from the sensed data, $\mathcal{P}$=$\{p_1, p_2, \cdots, p_N\}$. The mapping between the users and the parameters is represented as,

$$\mathcal{M}[x][y] = \begin{cases} 1, & \text{if user } u_x \text{ selects parameter } p_y \\ 0, & \text{otherwise} \end{cases} \tag{7.2}$$

where $0 \leq x \leq n$ and $0 \leq y \leq N$.

Further, the proposed system searches for users with similar types of needs at a particular zone with a similar selection of parameters from the above. In our approach, the similarity score decides the similar requirements of the users in neighboring vehicles. The likelihood score between the two registered users is determined based on the parameter selection of both users. The similarity score $S$ between the user and other neighboring users is mathematically represented as

$$S(\alpha) = \mathcal{L} \tag{7.3a}$$

$$\mathcal{L} = \Pi_{i=1}^{N} \left( \frac{\frac{p_o^i}{p_t^i}}{\frac{p_o^i sum}{p_t^i sum}} \right) \tag{7.3b}$$

where $p_o^i$ and $p_t^i$ are the selected decision parameters by other neighboring users and current users, whose decisions are being processed, respectively. We consider $\alpha$ to possess a positive value, which varies from 0 to 1. $\mathcal{L}$ represents the likelihood function by observing $N$ neighboring vehicles in each zone. For each zone, the delivered decision parameters are randomly distributed. In this work, we consider that the same SSP is working in multiple zones within the destination. However, certain hand-off mechanisms are applied for the continuation of service.

$$\begin{cases} if\alpha \geq \theta, & \text{same decision plan for the other users} \\ if\alpha \leq, & \text{new decision plan} \end{cases} \tag{7.4}$$

where $\theta$ is a threshold value decided by the SSP for each zone. $\theta$ is represented as-

$$\theta = f(z_i) \tag{7.5}$$

***Self-learning:*** The self-learning phase of the recommender system works on the various types of dynamic parameters with the help of fuzzy learning technology. The dynamic road and driver behavior parameters such as harsh braking, harsh acceleration, and max speed, while minimum temperature, maximum temperature, Humidity, and amount of rainfall act as the weather condition parameters. The scalar value of the parameters enters into the self-learning phase and the class of risk values is computed. In our fuzzy system, crisp values are fuzzified using the triangular fuzzification method. The risk value class is categorized into three classes $\{low_risk, medium_risk, high_risk\}$. We consider $x$, $y$, and $z$ as the linguistic variables which specify the different safety measures value such as road-specific, environment-specific, and driver-specific. Therefore, the fuzzy rule-based technique defines how the risk class is decided.

$$\text{If } x \text{ is } r_i,\ y \text{ is } w_i,\ z \text{ is } d_i, \text{ risk}(R_f) \text{ is } low_risk \tag{7.6a}$$

$$\text{If } x \text{ is } r_j,\ y \text{ is } w_j,\ z \text{ is } d_j, \text{ risk}(R_f) \text{ is } medium_risk \tag{7.6b}$$

$$\text{If } x \text{ is } r_k,\ y \text{ is } w_k,\ z \text{ is } d_k, \text{ risk}(R_f) \text{ is } high_risk \tag{7.6c}$$

From the above equations, the risk factor is classified and corresponding fuzzified risk membership values are generated according to the below equations.

$$\mu^{R_f}_{low_risk} = max(\mu_{X_i}, \mu_{Y_i}, \mu_{Z_i}) \tag{7.7a}$$

$$\mu^{R_f}_{medium_risk} = max(\mu_{X_j}, \mu_{Y_j}, \mu_{Z_j}) \tag{7.7b}$$

$$\mu^{R_f}_{high_risk} = max(\mu_{X_k}, \mu_{Y_k}, \mu_{Z_k} \tag{7.7c}$$

For example, triangular fuzzification method works as,

$$\mu_i(P) = \begin{cases} 0, & P \leq a, \\ \frac{P-a}{b-a}, & a \leq P \leq b, \\ \frac{c-P}{c-b}, & b \leq P \leq c, \\ 0, & c \leq P \end{cases}.$$

Each time the above risk factor ($R_f$) is trained whenever alteration in the corresponding safety measure value.

***Optimized Safety Decision Plan:***

Therefore, the new decision plan is generated and delivered to the users after analyzing the risk scores. In Equation 7.8, we generate the risk score after analyzing the parameters selected by users. The users select parameters whenever they require safety services during their journey. The decision is delivered by the SSP to the users. Therefore, the decision may vary from time to time, whenever new sensed data is transmitted to the system. Risk score($R_s$) per user is computed as-

$$R_s = f(\sum_{i=1}^{n} p_s^i \mod \theta) \tag{7.8}$$

Further, a safety decision plan is generated based on risk factor value. The decision plan may contain several pieces of information such as maintaining the speed limit, weather being bad after x km, harsh driving, and the suggestion of an alternate route. As a decision plan is generated based on the risk factor value, optimization takes place by minimizing the error between this risk factor value and the previously mentioned risk score for $k$ zones. Mathematically -

$$\mathcal{E} = \frac{(R_s - R_f)^2}{k} \tag{7.9}$$

We formulate the decision plan after optimizing the below function, which is represented as

$$\text{Minimize} \quad \mathcal{E} \tag{7.10}$$

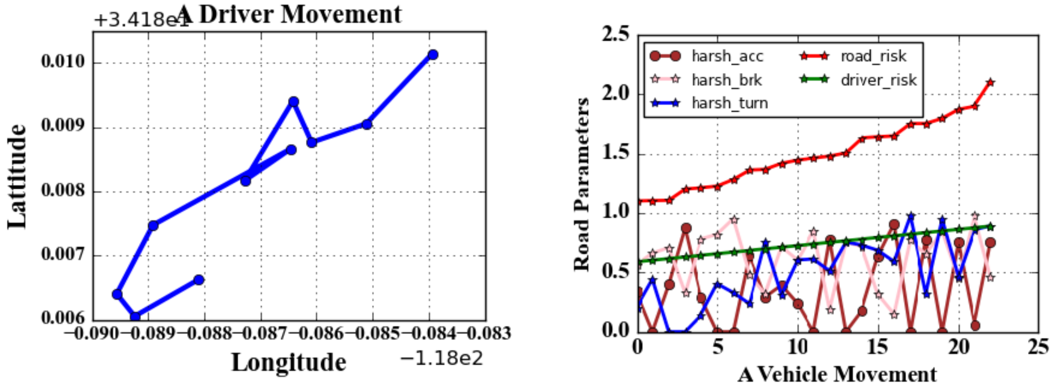
subject to $R_s \geq 0$, $R_f \geq 0$, $R_s \neq R_f$.

## 7.3 Performance Analysis

Performance analysis of the following parameters is determined in the Python emulator tool with a sample Kaggle dataset. The dataset contains more than 3000 samples on various types of attributes such as weather-specific, road condition-specific, driving specific. Our system analyzes the following parameters evaluates risk factors and delivers an optimized safety plan to the users.

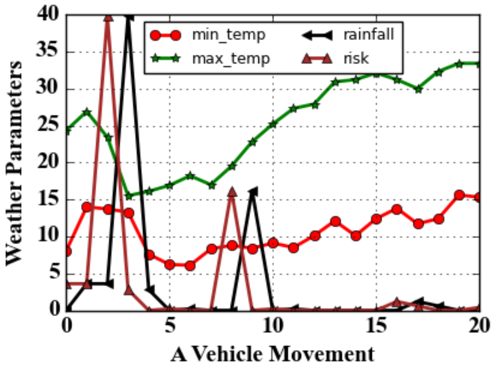
***Risk Analysis***: We consider different risk parameters such as weather, road condition, and driving behavior, and a dataset that contains the scalar values of these parameters. Risk analysis is an important factor in our recommendation system. The system analyzes the risk factors for each of the parameters. Additionally, this dataset contains the location-wise movement of the vehicle - longitude and latitude values. The sample driving movement of a driver is demonstrated in Fig. 7.3a.

**Road Condition:** We consider harsh breaking, harsh acceleration, and harsh turning as the parameters to calculate the risk factor due to road conditions. We take twenty-three driver samples to determine Cumulative Distribution Function (CDF) and then the overall risk is evaluated. The risk score for the specified road conditions is shown in Fig. 7.3b. We observe that harsh acceleration is responsible for the risk, though all are participating as the risk factors. Road risk is 50% higher than the driver risk score.



(a) A Driver Movement

(b) Risk on Road Conditions

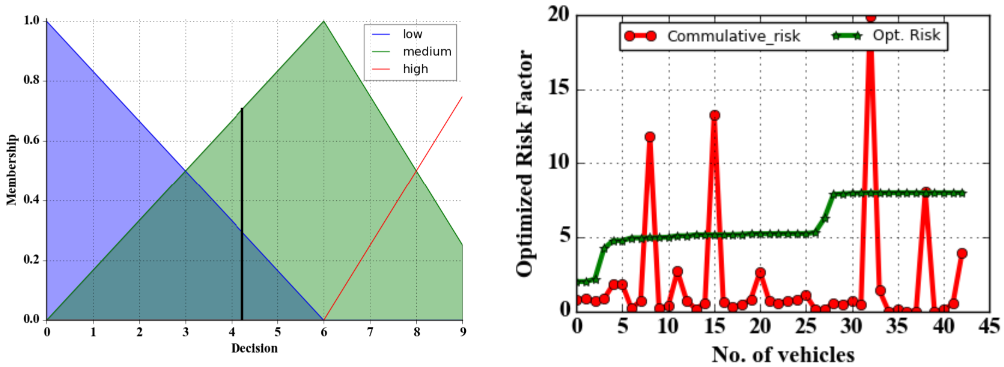


(c) Risk on Weather Conditions

FIGURE 7.3: A Driver Movement, and Risk Analysis

**Weather Condition:** The various weather condition parameters such as minimum temperature, maximum temperature, and rainfall, generate risk scores. We observe that the value of risk generated is quite high when there is heavy rainfall in specific locations. In Fig. 7.3c, the trend of risk values generated depends on the rainfall amount other than temperature values. The reason behind this is that heavy rainfall causes accidents and congestion.

***Decision Recommendation:*** Each of the parameters is fuzzified using our defined fuzzy logic technique. We generate seven types of membership values for the road, weather, and driving parameters. The different classes of membership value for the parameters are *dismal, poor, mediocre, average, decent, good, excellent*. We define ten rules in a fuzzy inference system to get fuzzified decisions. The fuzzified decision has three membership classes such as *low, medium, high*. We put the rule in such a manner that any worse value of the parameters may result in a risky situation. An example of a fuzzified decision is analyzed in Fig. 7.4a. In this case, the decision is in low to medium risk and a certain safety plan is delivered to the driver. Before providing the safety plan, the risk values are optimized with safety scores for driving to a particular location such that different risk measures or factors as in Fig. 7.4b.



(a) Example of Fuzzified Decision

(b) Optimized Risk Factor

FIGURE 7.4: Result Analysis

***Delay Incurred***: We calculate the total delay for our proposed approach. Introducing edge intelligence layer in Safe-aaS, delay is minimized upto 90.58% than traditional Safe-aaS. Fig. 7.5, delay1 is marked as the delay incurred for traditional Safe-aaS and delay2 for Safe-aaS with edge intelligence layer. We observe the increasing trend of delay in both cases in the increasing number of sensor nodes. The possible reason for this trend is as number of sensor nodes usage raises, more decisions are generated. As a result, the more total delay is incurred.

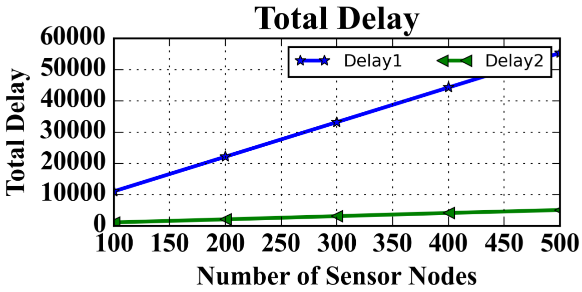


FIGURE 7.5: Delay Calculation

## 7.4 Conclusion

In this work, we proposed a recommendation system structured into three distinct phases: data preparation, self-learning, and decision plan generation. The goal of this system is to provide on-road vehicle drivers or owners with optimized safety plans, leveraging data-driven insights to enhance journey safety. We achieve this by dividing a driver's journey into several "operating zones," each managed under a single SSP, and by analyzing various risk factors within each zone.

**Data Preparation Phase:** In this initial phase, we gather and organize data essential for calculating a comprehensive safety score. This score incorporates various safety measures, including road conditions, weather conditions, and driver behavior. These factors are quantified to form a holistic risk profile for the driver's environment and actions.

**Self-Learning Phase:** In the second phase, we implement a fuzzy rule-based technique that "learns" from the dataset. By applying fuzzy logic, our system can interpret data with varying degrees of uncertainty and convert it into actionable risk values. This learning process enables our system to dynamically adjust to new data and understand complex patterns in road safety risks. Additionally, we calculate a similarity score to determine the relevance of safety plans from other neighboring users who share similar risk profiles. This score helps the system identify drivers who are facing comparable risk factors, enabling effective and context-aware sharing of safety plans.

**Decision Plan Generation Phase:** In the final phase, an optimization function generates a tailored decision plan for each user. This function aims to provide a decision plan that not only maximizes safety but also minimizes unnecessary interventions, offering a practical guide for a safe and smooth journey. Based on the generated decision plan, users receive actionable insights and recommendations, improving their awareness of potential risks and empowering them to take preemptive actions.

Overall, this recommendation system leverages a blend of data analytics, self-learning techniques, and optimization to deliver accurate, context-sensitive safety guidance. By continuously learning from real-world data and adjusting its recommendations, our system strives to enhance the quality of Safe-aaS, helping drivers navigate journeys with a greater sense of security and confidence.

In the future, we intend to focus on showcasing the cost efficiency of our system in a detailed manner. This presentation will highlight various aspects that contribute to its economical advantages. Additionally, we have plans in place to further enhance the features within our application interface, making it more user-friendly and intuitive. Alongside these improvements, we also aim to develop a more tailored safety plan that specifically addresses the potential risks of collisions and accidents, ensuring that our system prioritizes safety for all users involved.

# 8

# Conclusion and Future Scope

This thesis proposed and demonstrated an effective framework for pricing and resource management in Safe-aaS platforms, specifically designed to provide real-time safety decisions for on-road vehicles. The study addressed the technological and economic challenges associated with delivering accurate, timely, and cost-effective safety services in a dynamic vehicular environment. By integrating multiple technological layers and intelligent pricing mechanisms, this work provided a robust, adaptable solution for modern transportation systems.

## 8.1 Key Contributions and Findings

**Adaptive Decision-Generation Mechanism (Dec-Safe)**: A foundational element of this work is the development of the Dec-Safe mechanism, which enables Safe-aaS to make adaptive, efficient safety decisions by distinguishing between static and dynamic parameters. Static parameters, such as road conditions and geographical data, do not frequently change, allowing the system to conserve sensor and computational resources by avoiding redundant data collection. Conversely, dynamic parameters such as real-time traffic or weather updates are processed in real-time, ensuring that critical safety information is promptly delivered to users. This adaptive approach optimizes storage, reduces sensor fatigue, and minimizes decision delivery time, directly enhancing the overall efficiency and reliability of Safe-aaS platforms.

**QoS-Aware Pricing Model (Q-Safe):** Pricing in a service-oriented, real-time safety platform requires balancing cost efficiency for users and profitability for SSPs. The Q-Safe model addresses this need by offering an intelligent, differentiated pricing structure. It categorizes decision parameters into low- and high-cost based on their static or dynamic nature, while considering each user's demand and location-based needs. Through a game-theoretic formulation, Q-Safe optimizes pricing to maintain high QoS without inflating costs, thus ensuring that the system remains accessible to a broad user base. By incentivizing users to select efficient decision parameters, Q-Safe aligns user preferences with SSP goals, fostering sustainable growth of the Safe-aaS ecosystem.

**Introduction of Edge-Intelligence Layer:** Addressing latency is paramount in any real-time safety platform, especially one involving the immediate needs of on-road vehicles. The addition of an edge intelligence layer in this thesis distributes data processing closer to the point of data generation (i.e., vehicle or roadside sensors), reducing the cloud's processing load and lowering latency. Edge servers near data sources process time-sensitive information locally, thereby significantly minimizing the delay in decision generation. The edge intelligence layer also decreases the bandwidth requirements of Safe-aaS, enhancing scalability and efficiency as the number of on-road vehicles and data volume increases.

**Ultra-Low Delay Pricing Mechanism for Critical Safety Decisions:** A central challenge in Safe-aaS systems is ensuring ultra-low delay in delivering time-critical safety information, which can make the difference between a safe journey and a hazardous incident. To address this, the thesis proposes an ultra-low delay pricing mechanism that prioritizes quick decision delivery by leveraging edge resources and cloud resources selectively. For high-priority decisions, this mechanism dynamically adjusts pricing to ensure that latency-sensitive data processing is prioritized. This pricing structure provides an economic incentive for SSPs to prioritize high-impact, low-latency services, contributing to overall road safety.

**Development of a Safety Recommender System:** This thesis extends the Safe-aaS framework by developing a Safety Recommender System that provides personalized, real-time safety recommendations to users based on their selected routes, user-specific risk factors, and the broader traffic environment. By aggregating both static and dynamic parameters, the recommender system offers a customized safety profile to each user, which can also be shared with nearby vehicles. This approach enables a collaborative safety model, where multiple users can benefit from shared safety insights, thus enhancing the resilience and proactive safety of the entire vehicular network.

## 8.2 Implications of the Research

The proposed Safe-aaS framework's contributions are twofold, impacting both technological advancements and the business model of safety services in transportation:

**Technological Impact:** The integration of adaptive decision-making, edge computing, and QoS-aware pricing contributes to a more responsive, reliable, and resource-efficient safety platform. By leveraging AI and IoT at the edge level, this work paves the way for ultra-low latency applications that can operate at scale, even in congested urban environments. Additionally, the flexible decision-generation model introduced here could serve as a prototype for other time-critical IoT systems, providing a scalable architecture for various applications in healthcare, public safety, and IIoT.

**Economic and Operational Impact:** The Q-Safe pricing mechanism allows SSPs to maximize profitability without compromising service quality. As Safe-aaS platforms grow, they require a business model that balances high service quality with cost control for end-users. This work demonstrates a financially viable pathway for Safe-aaS, which can adapt dynamically to user demand and market fluctuations, ultimately ensuring that safe driving technology can become accessible to a broad range of vehicle owners and fleet operators.

## 8.3 Future Directions

**This research opened multiple avenues for future work in Safe-aaS and related fields:**

***Enhanced Machine Learning for Decision Customization:*** Future studies might explore advanced machine learning and predictive analytics techniques to further customize decision parameters based on user profiles, driving habits, or environmental conditions. Real-time learning algorithms could adapt safety recommendations based on user history, making Safe-aaS more responsive to individual driving styles and risk factors.

***Data Privacy and Security Enhancements:*** Safe-aaS platforms deal with sensitive user data, requiring robust privacy and security protocols to prevent unauthorized access or misuse. Future work could focus on implementing advanced cryptographic techniques to secure Safe-aaS data transmissions and storage without compromising performance.

***Scalability in Multi-Region Safe-aaS Deployments:*** As transportation networks expand, Safe-aaS platforms will need to operate across multiple regions and jurisdictions, each with unique regulatory and operational challenges. Expanding the framework to accommodate cross-region data sharing, localized pricing models, and region-specific risk factors will be essential in scaling Safe-aaS platforms to larger, global networks.

***Integration with Autonomous and Connected Vehicles:*** With the ongoing rise of autonomous vehicles, Safe-aaS systems must evolve to integrate seamlessly with autonomous driving platforms. Future research could investigate how Safe-aaS can enhance the safety of autonomous vehicles through real-time data sharing, predictive analytics, and adaptive responses to environmental changes.

***Exploring Blockchain for Decentralized Trust Management:*** Given the number of stakeholders in Safe-aaS—ranging from users and SSPs to sensor vendors and vehicle owners—decentralized trust mechanisms such as blockchain could enhance transparency, accountability, and transaction security. Future research could explore blockchain's role in managing data provenance, payment transactions, and decision validation in a decentralized Safe-aaS network.

In summary, this thesis provided a comprehensive framework for delivering real-time, user-centered safety services through Safe-aaS, with practical applications in the emerging IoT and connected transportation ecosystems. Through advanced resource management, adaptive decision-making, and intelligent pricing, this research created a foundation for a scalable, efficient, and sustainable approach to on-road safety, offering a substantial contribution to the future of intelligent transportation systems.

# References

[1] P. S. Saarika, K. Sandhya, and T. Sudha, "Smart transportation system using iot," in *2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon)*, pp. 1104–1107, 2017. 1

[2] P. Koopman and M. Wagner, "Autonomous vehicle safety: An interdisciplinary challenge," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 1, pp. 90–96, 2017. 1, 2

[3] C. Perera, C. H. Liu, and S. Jayawardena, "The emerging internet of things marketplace from an industrial perspective: A survey," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 4, pp. 585–598, 2015. 1

[4] Y. Liao, E. de Freitas Rocha Loures, and F. Deschamps, "Industrial internet of things: A systematic literature review and insights," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4515–4525, 2018. 1

[5] X. Mu and M. F. Antwi-Afari, "The applications of internet of things (iot) in industrial management: a science mapping review," *International Journal of Production Research*, vol. 62, no. 5, pp. 1928–1952, 2024. 1

[6] Y. Yu, H. Guan, and Z. Ji, "Automated detection of urban road manhole covers using mobile laser scanning data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3258–3269, 2015. 1, 2, 14

[7] M. Chen and Y. Wang, "On the key technologies of internet of vehicles and its innovative application of integration with 5g," in *2020 International Conference on Artificial Intelligence and Electromechanical Automation (AIEA)*, pp. 32–35, 2020. 1

[8] S. Latif, H. Afzaal, and N. A. Zafar, "Intelligent traffic monitoring and guidance system for smart city," in *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, pp. 1–6, 2018. 1

[9] M. R. Dey, S. Sharma, R. C. Shit, C. P. Meher, and H. K. Pati, "Iov based real-time smart traffic monitoring system for smart cities using augmented reality," in *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*, pp. 1–6, 2019. 1

[10] M. A. Rahman, M. S. Hossain, A. J. Showail, N. A. Alrajeh, and A. Ghoneim, "Ai-enabled iiot for live smart city event monitoring," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 2872–2880, 2023. 1

[11] C. Tang, C. Zhu, H. Wu, X. Wei, Q. Li, and J. J. P. C. Rodrigues, "A game theoretical pricing scheme for vehicles in vehicular edge computing," in *2020 16th International Conference on Mobility, Sensing and Networking (MSN)*, pp. 17–22, 2020. 1, 20, 22

[12] A. A. Afify and B. Mokhtar, "Machine learning-based services provisioning for intelligent internet of vehicles," in *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*, pp. 51–54, 2021. 2

[13] L.-C. Wang, H. Gačanin, D. Niyato, Y.-J. Chen, C.-H. Liu, and A. Anpalagan, "Artificial intelligence for autonomous vehicular communication networks [from the guest editors]," *IEEE Vehicular Technology Magazine*, vol. 17, no. 2, pp. 83–84, 2022. 2

[14] F. Falahatraftar, S. Pierre, and S. Chamberland, "A centralized and dynamic network congestion classification approach for heterogeneous vehicular networks," *IEEE Access*, vol. 9, pp. 122284–122298, 2021. 2

[15] U. Budak, V. Bajaj, Y. Akbulut, O. Atila, and A. Sengur, "An effective hybrid model for eeg-based drowsiness detection," *IEEE Sensors Journal*, vol. 19, no. 17, pp. 7624–7631, 2019. 2

[16] Y. Xing, C. Lv, Z. Zhang, H. Wang, X. Na, D. Cao, E. Velenis, and F.-Y. Wang, "Identification and analysis of driver postures for in-vehicle driving activities and secondary tasks recognition," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 1, pp. 95–108, 2018. 2

[17] C. Wang, Q. Sun, Y. Guo, R. Fu, and W. Yuan, "Improving the user acceptability of advanced driver assistance systems based on different driving styles: A case study of lane change warning systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 10, pp. 4196–4208, 2020. 2

[18] J. Prakash, L. Murali, N. Manikandan, N. Nagaprasad, and K. Ramaswamy, "A vehicular network based intelligent transport system for smart cities using machine learning algorithms," *Scientific Reports*, vol. 14, 2024. 2

[19] C. Roy, A. Roy, S. Misra, and J. Maiti, "Safe-aaS: Decision Virtualization for Effecting Safety-as-a-Service," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1690–1697, 2018. 2, 19, 22, 36, 45, 63, 67

[20] P. K. Deb, C. Roy, A. Roy, and S. Misra, "Deft: Decentralized multiuser computation offloading in a fog-enabled iov environment," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 15978–15987, 2020. 5

[21] C. Roy, S. Misra, J. Maiti, and U. Chakravarty, "Safe-Serv: Energy-Efficient Decision Delivery for Provisioning Safety-as-a-Service," *IEEE Transactions on Services Computing*, vol. 15, pp. 1954–1966, july 2022. 5

[22] C. Roy, C. R. Chowdhury, S. Misra, and J. Maiti, "Dq-map: Dynamic decision query mapping for provisioning safety-as-a-service in iot," *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 3150–3157, 2022. 5

[23] C. Roy and S. Misra, “Safe-Passé: Dynamic Handoff Scheme for Provisioning Safety-as-a-Service in 5G-Enabled Intelligent Transportation System,” *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2021. 5

[24] P. Pradhan, C. Roy, S. Misra, and S. Chattopadhyay, “Dec-safe: Dynamic decision generation mechanism for delivering safety services in vehicular networks,” *IEEE Transactions on Vehicular Technology*, vol. 72, no. 12, pp. 15280–15289, 2023. 10

[25] P. Pradhan, C. Roy, and S. Misra, “Q-safe: Qos-aware pricing scheme for provisioning safety-as-a-service,” *IEEE Transactions on Services Computing*, vol. 16, no. 1, pp. 515–524, 2023. 10

[26] P. Pradhan, C. Roy, S. Misra, and S. Chattopadhyay, “Edge intelligence-based safety-as-a-service platform for social iov environment,” in *ICC 2023 - IEEE International Conference on Communications*, pp. 4943–4948, 2023. 10

[27] J. Zhang, B. Chen, Y. Zhao, X. Cheng, and F. Hu, “Data security and privacy-preserving in edge computing paradigm: Survey and open issues,” *IEEE Access*, vol. 6, pp. 18209–18237, 2018. 14, 18

[28] V. Rishiwal and H. Khan, “Automatic pothole and speed breaker detection using android system,” in *2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 1270–1273, 2016. 14

[29] Y. Ikeda and M. Inoue, “An estimation of road surface conditions using participatory sensing,” in *2018 International Conference on Electronics, Information, and Communication (ICEIC)*, pp. 1–3, 2018. 14

[30] T. Satoh, A. Hiromori, H. Yamaguchi, and T. Higashino, “A novel estimation method of road condition for pedestrian navigation,” in *2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, pp. 427–432, 2015. 14

[31] H. Cheng, N. Zheng, X. Zhang, J. Qin, and H. van de Wetering, “Interactive road situation analysis for driver assistance and safety warning systems: Framework and algorithms,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 1, pp. 157–167, 2007. 14

[32] Y. Du, C. Liu, Y. Song, Y. Li, and Y. Shen, “Rapid estimation of road friction for anti-skid autonomous driving,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2461–2470, 2020. 14

[33] A. Gregoriades, A. Sutcliffe, G. Papageorgiou, and P. Louvieris, “Human-centered safety analysis of prospective road designs,” *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 40, no. 2, pp. 236–250, 2010. 14

[34] M. Chaturvedi and S. Srivastava, “Multi-modal design of an intelligent transportation system,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 8, pp. 2017–2027, 2017. 15

[35] H. Bar-Gera, "Evaluation of a cellular phone-based system for measurements of traffic speeds and travel times: A case study from israel," *Transportation Research Part C: Emerging Technologies*, vol. 15, no. 6, pp. 380–391, 2007. 15

[36] N. Madrid and P. Hurtik, "Lane departure warning for mobile devices based on a fuzzy representation of images," *Fuzzy Sets and Systems*, vol. 291, pp. 144–159, 2016. Special Issue on Selected Papers from FSTA 2014. 15

[37] Z. Huang, "Research on lane departure avoidance system of high-speed vehicle," *Journal of Mechanical Engineering*, vol. 49, p. 157, 01 2013. 15

[38] V. K. Kukkala, J. Tunnell, S. Pasricha, and T. Bradley, "Advanced driver-assistance systems: A path toward autonomous vehicles," *IEEE Consumer Electronics Magazine*, vol. 7, no. 5, pp. 18–25, 2018. 15

[39] M. Fazeen, B. Gozick, R. Dantu, M. Bhukhiya, and M. C. González, "Safe driving using mobile phones," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1462–1468, 2012. 15

[40] S. Fernandez and T. Ito, "Driver classification for intelligent transportation systems using fuzzy logic," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1212–1216, 2016. 15, 91

[41] S. Kaplan, M. A. Guvensan, A. G. Yavuz, and Y. Karalurt, "Driver behavior analysis for safe driving: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3017–3032, 2015. 15, 18, 91

[42] C. Marina Martinez, M. Heucke, F.-Y. Wang, B. Gao, and D. Cao, "Driving style recognition for intelligent vehicle control and advanced driver assistance: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 666–676, 2018. 15

[43] K. T. Chui, K. F. Tsang, H. R. Chi, B. W. K. Ling, and C. K. Wu, "An accurate ecg-based transportation safety drowsiness detection scheme," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 4, pp. 1438–1452, 2016. 15

[44] M. Agarwal, T. Maze, and R. Souleyrette, "Impact of weather on urban freeway traffic low characteristics and facility capacity," 09 2005. 15

[45] D. Pavlou, G. Christodoulou, and G. Yannis, "The impact of weather conditions and driver characteristics on road safety on rural roads," *Transportation Research Procedia*, vol. 72, pp. 4081–4088, 2023. TRA Lisbon 2022 Conference Proceedings Transport Research Arena (TRA Lisbon 2022),14th-17th November 2022, Lisboa, Portugal. 15

[46] C. Chen, X. Zhao, H. Liu, G. Ren, and X. Liu, "Influence of adverse weather on drivers' perceived risk during car following based on driving simulations," *Journal of Modern Transportation*, vol. 27, 09 2019. 15, 18

[47] L. Qi, "Research on intelligent transportation system technologies and applications," Aug 2008. 16

[48] O. Kaiwartya, A. H. Abdullah, Y. Cao, A. Altameem, M. Prasad, C.-T. Lin, and X. Liu, “Internet of vehicles: Motivation, layered architecture, network model, challenges, and future aspects,” *IEEE Access*, vol. 4, pp. 5356–5373, 2016. 16

[49] S. Mumtaz, K. M. Saidul Huq, M. I. Ashraf, J. Rodriguez, V. Monteiro, and C. Politis, “Cognitive vehicular communication for 5g,” *IEEE Communications Magazine*, vol. 53, no. 7, pp. 109–117, 2015. 16

[50] S.-h. An, B.-H. Lee, and D.-R. Shin, “A survey of intelligent transportation systems,” in *2011 Third International Conference on Computational Intelligence, Communication Systems and Networks*, pp. 332–337, 2011. 16

[51] M. Abuelela and S. Olariu, “Taking vanet to the clouds,” in *Proceedings of the 8th International Conference on Advances in Mobile Computing and Multimedia*, MoMM ’10, (New York, NY, USA), p. 6–13, Association for Computing Machinery, 2010. 17

[52] A. Moujahid, M. ElAraki Tantaoui, M. D. Hina, A. Soukane, A. Ortalda, A. ElKhadimi, and A. Ramdane-Cherif, “Machine learning techniques in adas: A review,” in *2018 International Conference on Advances in Computing and Communication Engineering (ICACCE)*, pp. 235–242, 2018. 17

[53] B. K. M, A. Basit, K. MB, G. R, and K. SM, “Road accident detection using machine learning,” in *2021 International Conference on System, Computation, Automation and Networking (ICSCAN)*, pp. 1–5, 2021. 17

[54] S. Bhattacharya, H. Jha, and R. Nanda, “Application of iot and ai in road safety.,” 03 2022. 17

[55] S. Alagarsamy, P. Nagaraj, B. Srikanth, C. V. Krishna, G. Bharath, and S. S. Kalyan, “A novel machine learning technique for predicting road accidents,” in *2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)*, pp. 1547–1551, 2023. 17

[56] N. Dogru and A. Subasi, “Traffic accident detection using random forest classifier,” in *2018 15th Learning and Technology Conference (LT)*, pp. 40–45, 2018. 17

[57] T. Lillicrap, J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, “Continuous control with deep reinforcement learning,” *CoRR*, 09 2015. 17

[58] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, “A tutorial on uavs for wireless networks: Applications, challenges, and open problems,” *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2334–2360, 2019. 17

[59] K. P. Seng, L. M. Ang, and E. Ngharamike, “Artificial intelligence internet of things: A new paradigm of distributed sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 18, no. 3, p. 15501477211062835, 2022. 17

[60] U. Khadam, P. Davidsson, and R. Spalazzese, “Exploring the role of artificial intelligence in internet of things systems: A systematic mapping study,” *Sensors*, vol. 24, no. 20, 2024. 17

[61] K. Banerjee, V. Bali, A. Sharma, D. Aggarwal, A. Yadav, A. Shukla, and P. Srivastav, “Traffic accident risk prediction using machine learning,” in *2022 International Mobile and Embedded Technology Conference (MECON)*, pp. 76–82, 2022. 17

[62] T. Augustine and S. Shukla, “Road accident prediction using machine learning approaches,” in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, pp. 808–811, 2022. 17

[63] F. Tango and M. Botta, “Real-time detection system of driver distraction using machine learning,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 894–905, 2013. 18

[64] J. Kim, “A study on the development of traffic safety risk information sharing technology through vehicle-road cooperation,” in *2021 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–3, 2021. 18

[65] H. Zhao, J. Zhang, X. Li, Q. Wang, and H. Zhu, “Deep learning-based prediction of traffic accident risk in vehicular networks,” in *2020 IEEE Globecom Workshops (GC Wkshps*, pp. 1–5, 2020. 18

[66] S. Bitam and A. Mellouk, “Its-cloud: Cloud computing for intelligent transportation system,” in *2012 IEEE Global Communications Conference (GLOBECOM)*, pp. 2054–2059, 2012. 18

[67] P. Jaworski, T. Edwards, J. Moore, and K. Burnham, “Cloud computing concept for intelligent transportation systems,” in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 391–936, 2011. 18

[68] W. He, G. Yan, and L. D. Xu, “Developing vehicular data cloud services in the iot environment,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1587–1595, 2014. 18

[69] J. Zhao, Q. Li, Y. Gong, and K. Zhang, “Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7944–7956, 2019. 19

[70] A. Ahmed, D. Abdullah, S. Iftikhar, I. Ahmad, S. Ajmal, and Q. Hussain, “A novel blockchain based secured and qos aware iot vehicular network in edge cloud computing,” *IEEE Access*, vol. 10, pp. 1–1, 01 2022. 19

[71] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, “A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications,” *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1125–1142, 2017. 19

[72] A. A. Khan, M. H. Rehmani, and A. Rachedi, “Cognitive-radio-based internet of things: Applications, architectures, spectrum related functionalities, and future research directions,” *IEEE Wireless Communications*, vol. 24, pp. 17–25, 2017. 19

[73] M. Noor-A-Rahim, Z. Liu, H. Lee, M. O. Khyam, J. He, D. Pesch, K. Moessner, W. Saad, and H. V. Poor, “6g for vehicle-to-everything (v2x) communications: Enabling

technologies, challenges, and opportunities," *Proceedings of the IEEE*, vol. 110, no. 6, pp. 712–734, 2022. 19

[74] S. Adnan Yusuf, A. Khan, and R. Souissi, "Vehicle-to-everything (v2x) in the autonomous vehicles domain – a technical review of communication, sensor, and ai technologies for road user safety," *Transportation Research Interdisciplinary Perspectives*, vol. 23, p. 100980, 2024. 19

[75] F. Arena, G. Pau, and A. Severino, "V2x communications applied to safety of pedestrians and vehicles," *Journal of Sensor and Actuator Networks*, vol. 9, no. 1, 2020. 19

[76] M. Dabbagh, B. Hamdaoui, M. Guizani, and A. Rayes, "Exploiting task elasticity and price heterogeneity for maximizing cloud computing profits," *IEEE Transactions on Emerging Topics in Computing*, vol. 6, no. 1, pp. 85–96, 2018. 20

[77] C. Roy, S. Misra, J. J. P. C. Rodrigues, and U. Chakravarty, "RegPrice: Region-Based Pricing Scheme for Provisioning Safety-as-a-Service in IoT Applications," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3017–3026, 2021. 20, 46, 55

[78] N. Semret, R. Liao, A. Campbell, and A. Lazar, "Pricing, provisioning and peering: Dynamic markets for differentiated internet services and implications for network interconnections," *Selected Areas in Communications, IEEE Journal on*, vol. 18, pp. 2499 – 2513, 01 2001. 20

[79] S.-H. Kim, S. Park, M. Chen, and C.-H. Youn, "An optimal pricing scheme for the energy-efficient mobile edge computation offloading with ofdma," *IEEE Communications Letters*, vol. 22, no. 9, pp. 1922–1925, 2018. 20

[80] J. M. Alvarez, T. Gevers, F. Diego, and A. M. Lopez, "Road Geometry Classification by Adaptive Shape Models," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 459–468, 2013. 22

[81] J. M. Álvarez, A. M. López, T. Gevers, and F. Lumbreras, "Combining priors, appearance, and context for road detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 3, pp. 1168–1178, 2014. 22

[82] J.-P. Jodoin, G.-A. Bilodeau, and N. Saunier, "Tracking All Road Users at Multimodal Urban Traffic Intersections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 11, pp. 3241–3251, 2016. 22

[83] C. Roy, S. Misra, J. Maiti, and U. Chakravarty, "Safe-Serv: Energy-Efficient Decision Delivery for Provisioning Safety-as-a-Service," *IEEE Transactions on Services Computing*, pp. 1–1, 2020. 22, 45, 63, 76

[84] C. Roy, S. Misra, and S. Pal, "Blockchain-Enabled Safety-as-a-Service for Industrial IoT Applications," *IEEE Internet of Things Magazine*, vol. 3, no. 2, pp. 19–23, 2020. 22

[85] P. Pradhan, C. Roy, and S. Misra, "Q-safe: Qos-aware pricing scheme for provisioning safety-as-a-service," *IEEE Transactions on Services Computing*, vol. 16, no. 1, pp. 515–524, 2023. 22

[86] L. Pozueco, N. Gupta, X. G. Pañeda, R. Garcia, A. G. Tuero, D. Melendi, A. Rionda, and V. Corcoba, "Analysis of Driving Patterns and On-Board Feedback-Based Training for Proactive Road Safety Monitoring," *IEEE Transactions on Human-Machine Systems*, vol. 50, no. 6, pp. 529–537, 2020. 22, 23

[87] A. Alsarhan, A. Y. Al-Dubai, G. Min, A. Y. Zomaya, and M. Bsoul, "A New Spectrum Management Scheme for Road Safety in Smart Cities," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 11, pp. 3496–3506, 2018. 22, 23

[88] Y. Yu, H. Guan, and Z. Ji, "Automated detection of urban road manhole covers using mobile laser scanning data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3258–3269, 2015. 22, 23

[89] R. Tian, L. Li, M. Chen, Y. Chen, and G. J. Witt, "Studying the Effects of Driver Distraction and Traffic Density on the Probability of Crash and Near-Crash Events in Naturalistic Driving Environment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1547–1555, 2013. 22

[90] B. R, M. J. R. G, H. M M, and K. K. K, "Vanet based accident alerting system," in *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 661–668, 2021. 36, 63, 76, 85

[91] S. Na, L. Xumin, and G. Yong, "Research on k-means clustering algorithm: An improved k-means clustering algorithm," in *2010 Third International Symposium on Intelligent Information Technology and Security Informatics*, pp. 63–67, 2010. 36

[92] K. Peng, V. C. M. Leung, and Q. Huang, "Clustering approach based on mini batch kmeans for intrusion detection system over big data," *IEEE Access*, vol. 6, pp. 11897–11906, 2018. 36

[93] E. Esenturk, A. G. Wallace, S. Khastgir, and P. Jennings, "Identification of traffic accident patterns via cluster analysis and test scenario development for autonomous vehicles," *IEEE Access*, vol. 10, pp. 6660–6675, 2022. 36

[94] Y. Qiu and P. Marbach, "Bandwidth allocation in ad hoc networks: a price-based approach," in $22^{nd}$ *Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No.03CH37428)*, vol. 2, pp. 797–807 vol.2, 2003. 44

[95] N. Semret, R. Liao, A. Campbell, and A. Lazar, "Pricing, Provisioning and Peering: Dynamic Markets for Differentiated Internet Services and Implications for Network Interconnections," *Selected Areas in Communications, IEEE Journal on*, vol. 18, pp. 2499 – 2513, 01 2001. 44

[96] S. Chaisiri, B. Lee, and D. Niyato, "Optimization of Resource Provisioning Cost in Cloud Computing," *IEEE Transactions on Services Computing*, vol. 5, no. 2, pp. 164–177, 2012. 44

[97] L. Guijarro, V. Pla, J. R. Vidal, and M. Naldi, "Maximum-Profit Two-Sided Pricing in Service Platforms Based on Wireless Sensor Networks," *IEEE Wireless Communications Letters*, vol. 5, no. 1, pp. 8–11, 2016. 44, 46, 55, 85

[98] M. M. Hassan and A. Alsanad, “Resource provisioning for cloud-assisted software defined wireless sensor network,” *IEEE Sensors Journal*, vol. 16, no. 20, pp. 7401–7408, 2016. 44

[99] A. Chakraborty, A. Mondal, A. Roy, and S. Misra, “Dynamic Trust Enforcing Pricing Scheme for Sensors-as-a-Service in Sensor-Cloud Infrastructure,” *IEEE Transactions on Services Computing*, pp. 1–1, 2018. 44, 45

[100] A. Roy, S. Misra, and P. Dutta, “Dynamic Pricing for Sensor-Cloud Platform in the Presence of Dumb Nodes,” *IEEE Transactions on Cloud Computing*, pp. 1–1, 2019. 44, 76

[101] A. Chakraborty, S. Misra, and A. Mondal, “Qos-aware dynamic cost management scheme for sensors-as-a-service,” *IEEE Transactions on Services Computing*, pp. 1–1, 2020. 44, 45

[102] T. Tettamanti, Török, and I. Varga, “Dynamic road pricing for optimal traffic flow management by using non-linear model predictive control,” *IET Intelligent Transport Systems*, vol. 13, no. 7, pp. 1139–1147, 2019. 44, 45

[103] P. Chavali and A. Nehorai, “Managing Multi-Modal Sensor Networks Using Price Theory,” *IEEE Transactions on Signal Processing*, vol. 60, no. 9, pp. 4874–4887, 2012. 44

[104] B. Baek, J. Lee, Y. Peng, and S. Park, “Three Dynamic Pricing Schemes for Resource Allocation of Edge Computing for IoT Environment,” *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4292–4303, 2020. 44, 45

[105] Z. Xiong, D. Niyato, P. Wang, Z. Han, and Y. Zhang, “Dynamic Pricing for Revenue Maximization in Mobile Social Data Market With Network Effects,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 1722–1737, 2020. 44, 45

[106] D. Lučanin, I. Pietri, S. Holmbacka, I. Brandic, J. Lilius, and R. Sakellariou, “Performance-Based Pricing in Multi-Core Geo-Distributed Cloud Computing,” *IEEE Transactions on Cloud Computing*, vol. 8, no. 4, pp. 1079–1092, 2020. 45

[107] D. Ardagna, B. Panicucci, and M. Passacantando, “Generalized Nash Equilibria for the Service Provisioning Problem in Cloud Systems,” *IEEE Transactions on Services Computing*, vol. 6, no. 4, pp. 429–442, 2013. 45

[108] A. Roy, S. Misra, and S. Nag, “PRIME: An Optimal Pricing Scheme for Mobile Sensors-as-a-Service,” *IEEE Transactions on Mobile Computing*, pp. 1–1, 2020. 45, 46, 51, 55, 85

[109] J. Guerreiro, L. Rodrigues, and N. Correia, “Resource allocation model for sensor clouds under the sensing as a service paradigm,” *Computers*, vol. 8, p. 18, 02 2019. 45

[110] S. Bouchelaghem and M. Omar, “Reliable and secure distributed smart road pricing system for smart cities,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1592–1603, 2019. 45

[111] M. Fallgren, M. Dillinger, J. Alonso-Zarate, M. Boban, T. Abbas, K. Manolakis, T. Mahmoodi, T. Svensson, A. Laya, and R. Vilalta, "Fifth-Generation Technologies for the Connected Car: Capable Systems for Vehicle-to-Anything Communications," *IEEE Vehicular Technology Magazine*, vol. 13, no. 3, pp. 28–38, 2018. 63

[112] J. Lin, W. Yu, X. Yang, P. Zhao, H. Zhang, and W. Zhao, "An Edge Computing Based Public Vehicle System for Smart Transportation," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 12635–12651, 2020. 63

[113] M. Boban, A. Kousaridas, K. Manolakis, J. Eichinger, and W. Xu, "Connected Roads of the Future: Use Cases, Requirements, and Design Considerations for Vehicle-to-Everything Communications," *IEEE Vehicular Technology Magazine*, vol. 13, no. 3, pp. 110–123, 2018. 63

[114] S. Cafiso, G. La Cava, and V. Cutello, "A fuzzy model for road accidents analysis," in *18th International Conference of the North American Fuzzy Information Processing Society - NAFIPS (Cat. No.99TH8397)*, pp. 139–143, 1999. 63

[115] A. Ali, S. Ud-Din, S. Saad, S. Ammad, K. Rasheed, and F. Ahmad, "Artificial neural network approach to study the effect of driver characteristics on road traffic accidents," in *International Conference on Data Analytics for Business and Industry (ICDABI)*, pp. 277–280, 2021. 63

[116] R. Izquierdo, I. Parra, J. Muñoz-Bulnes, D. Fernández-Llorca, and M. A. Sotelo, "Vehicle trajectory and lane change prediction using ann and svm classifiers," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1–6, 2017. 63

[117] S. Ranadheera, S. Maghsudi, and E. Hossain, "Computation Offloading and Activation of Mobile Edge Computing Servers: A Minority Game," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 688–691, 2018. 63

[118] K. Cao, L. Li, Y. Cui, T. Wei, and S. Hu, "Exploring Placement of Heterogeneous Edge Servers for Response Time Minimization in Mobile Edge-Cloud Computing," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 1, pp. 494–503, 2021. 64

[119] M. Abuelela and S. Olariu, "Taking VANET to the Clouds," in *Proceedings of the 8th International Conference on Advances in Mobile Computing and Multimedia*, MoMM '10, (New York, NY, USA), p. 6–13, Association for Computing Machinery, 2010. 75

[120] E. Meissner, T. Chantem, and K. Heaslip, "Optimizing Departures of Automated Vehicles From Highways While Maintaining Mainline Capacity," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3498–3511, 2016. 75

[121] Y. Du, C. Liu, Y. Song, Y. Li, and Y. Shen, "Rapid Estimation of Road Friction for Anti-Skid Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2461–2470, 2020. 76

[122] P. Pradhan, C. Roy, and S. Misra, "Q-safe: Qos-aware pricing scheme for provisioning safety-as-a-service," *IEEE Transactions on Services Computing*, vol. 16, no. 1, pp. 515–524, 2023. 76

[123] C. Roy, S. Misra, J. Maiti, and F. Nait-Abdesselam, “Diff-price: Differential pricing scheme for provisioning safety-as-a-service in vehicular iot applications,” *IEEE Transactions on Vehicular Technology*, vol. 71, no. 8, pp. 8189–8198, 2022. 76

[124] H. Shah-Mansouri, V. W. S. Wong, and R. Schober, “Joint Optimal Pricing and Task Scheduling in Mobile Cloud Computing Systems,” *IEEE Transactions on Wireless Communications*, vol. 16, no. 8, pp. 5218–5232, 2017. 76

[125] C. Roy, S. Misra, J. Maiti, and M. S. Obaidat, “DENSE: Dynamic Edge Node Selection for Safety-as-a-Service,” in *IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, 2019. 76

[126] F. Murtagh, “Multilayer perceptrons for classification and regression,” *Neurocomputing*, vol. 2, no. 5, pp. 183–197, 1991. 80

[127] M. Pecht, A. Ramakrishnan, J. Fazio, and C. Nash, “The Role of the U.S National Highway Traffic Safety Administration in Automotive Electronics Reliability and Safety Assessment,” *IEEE Transactions on Components and Packaging Technologies*, vol. 28, no. 3, pp. 571–580, 2005. 91

[128] A. Sharma and R. B. Battula, “FOOTREST: Safety on Roads Through Intelligent Transportation System,” in *International Conference on Information Networking (ICOIN)*, pp. 818–820, 2020. 91

[129] J. Caban, R. Karpiński, and D. Barta, “Road traffic accident injuries — Causes and biomaterial related treatment,” in *2018 XI International Science-Technical Conference Automotive Safety*, pp. 1–7, 2018. 91

[130] C. Marina Martinez, M. Heucke, F.-Y. Wang, B. Gao, and D. Cao, “Driving Style Recognition for Intelligent Vehicle Control and Advanced Driver Assistance: A Survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 666–676, 2018. 91

[131] K. C. Dey, A. Mishra, and M. Chowdhury, “Potential of Intelligent Transportation Systems in Mitigating Adverse Weather Impacts on Road Mobility: A Review,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1107–1119, 2015. 92

[132] M. Cools, E. Moons, and G. Wets, “Assessing the impact of weather on traffic intensity,” *Weather, Climate, and Society*, vol. 2, no. 1, pp. 60 – 68, 2010. 92

[133] A. Nasser and V. Simon, “A Novel Method for Analyzing Weather Effect on Smart City Traffic,” in *2021 IEEE 22nd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 335–340, 2021. 92

[134] S. Saranyadevi, R. Murugeswari, and S. Bathrinath, “Fuzzy Logic based Recommendation System for Road Accident Data Analysis,” in *IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*, pp. 1–6, 2019. 92

[135] S. Amri, M. Naoum, M. Lazaar, and M. A. Achhab, "Performing of users' road safety at intelligent transportation systems," 2020. 92

Samiran Chattopadhyay

PROFESSOR
Deptt. of Information Technology
JADAVPUR UNIVERSITY
Block -LB, Plot-8, Sector-3
Salt Lake, Kolkata-700106, India

Sudip Misra

Pabali Pradhan

**Sudip** Misra
प्राध्यापक /Professor
संगणक विज्ञान एवं अभियांत्रिकी विभाग
Computer Sc. & Engg. Deptt.
भा प्रौ सं खड़गपुर/IIT Khargapur