A federated learning model for integrating sustainable routing with the Internet of Vehicular Things using genetic algorithm

Article *in* Decision Analytics Journal · May 2024

DOI: 10.1016/j.dajour.2024.100486

5 authors, including:

Sushovan Khatua
Maulana Abul Kalam Azad University of Technology, West Bengal
6 PUBLICATIONS **10** CITATIONS
SEE PROFILE

Debashis De
Maulana Abul Kalam Azad University of Technology, West Bengal
700 PUBLICATIONS **9,271** CITATIONS
SEE PROFILE

Somnath Maji
Maulana abul kalam azad university of techonology
9 PUBLICATIONS **24** CITATIONS
SEE PROFILE

Samir Maity
Aalborg University
36 PUBLICATIONS **390** CITATIONS
SEE PROFILE

Decision Analytics Journal 11 (2024) 100486

Contents lists available at ScienceDirect

Decision Analytics Journal

journal homepage: www.elsevier.com/locate/dajour

A federated learning model for integrating sustainable routing with the Internet of Vehicular Things using genetic algorithm

Sushovan Khatua [a,*], Debashis De [a], Somnath Maji [a], Samir Maity [b], Izabela Ewa Nielsen [b]

[a] *Department of Computer Science and Engineering, Maulana Abul Kalam Azad University of Technology, NH-12, Haringhata, Nadia, West Bengal, 741249, India*
[b] *Operations Research Group, Department of Materials and Production, Aalborg University, Aalborg, 9220, Denmark*

A R T I C L E I N F O

Keywords:
Sustainable development goals
Federated learning
Internet of Vehicular Things
Sustainable routing
Vehicular networks
Genetic algorithms

A B S T R A C T

A distributed machine learning technique called federated learning allows numerous Internet of Things (IoT) edge devices to work together to train a model without sharing their raw data. Internet of Vehicular Things (IoVT) are an important tool in smart cities for moving objects, such as knowing the traffic patterns, road conditions, vehicle behavior, etc. To enhance traffic management and optimize routes, federated learning, and IoT must work jointly, which may achieve sustainable development goals (SDG) in many ways. This research outlines a system for federated learning in vehicular networks in smart cities. The suggested architecture considers the difficulties presented by such situations' restricted network connectivity, privacy issues, and security concerns. The framework employs a hybrid methodology integrating federated learning on a centralized server with local training on individual cars. The proposed framework is assessed based on a real-world dataset from a smart city through IoT devices. The findings demonstrate that the suggested method successfully increases model accuracy while preserving the confidentiality and security of the data. In this investigation, we incorporated the Federated Learning model, which can fetch all the information between arbitrary nodes and derive the Traffic, Fuel Cost, Safety, Parking Cost, and Transportation cost for a better routing approach. The suggested framework can be utilized to increase the effectiveness of the transportation system, decrease congestion in smart cities, and improve traffic management. We employ an improved genetic algorithm (iGA) with generation-dependent even mutation to tackle the emission in the smart environment.

1. Introduction

1.1. Motivation

Today, the transportation system is a crucial aspect of any city, and its challenges are increasing with the rapid growth of urbanization.[1] Smart cities leverage various technologies to improve transportation management and enhance the quality of life for citizens. [Internet of Things] (IoT) devices produced vast amounts of data, and vehicular networks were used to optimize the transportation sector and reduce congestion, optimize route plans, improve safety, etc. However, after getting the different data and information, these contain some sensitive information, and sharing it poses privacy and security concerns.

Federated learning (FL) is a promising solution to address the privacy and security concerns of sharing data in vehicular networks. It enables training machine learning models on data distributed across multiple edge/IoT devices without transferring the raw data to a central server. This approach preserves the privacy and security of the data while allowing for the training of accurate models. Motivated by the potential benefits of FL in vehicular networks, we propose a framework that utilizes this technique to improve traffic management, route planning, safety, parking, etc., in smart cities. The proposed framework addresses the challenges of limited network connectivity, privacy, and security concerns associated with collecting and sharing vehicular network data. The framework leverages a hybrid approach that combines local training on individual vehicles with FL on a centralized server. The proposed framework can potentially improve traffic management in smart cities, reduce congestion, and enhance the overall efficiency of the transportation system. It can also help address the privacy and security concerns of sharing vehicular network data. Our motivation for this research is to provide a practical and effective solution to optimize transportation management in smart cities while preserving the privacy and security of the data. Smart cities are becoming increasingly popular worldwide, where various technologies are deployed to improve urban services and enhance the quality of life for citizens. Managing the transportation system is one of the most significant challenges in smart

* Corresponding author.
E-mail addresses: sushovankhatua79@gmail.com (S. Khatua), dr.debashis.de@gmail.com (D. De), somnathmajivucs@gmail.com (S. Maji), maitysamir13@gmail.com (S. Maity), izabela@mp.aau.dk (I.E. Nielsen).

[1] https://www.arup.com/perspectives/publications/research/section/growing-smart-cities-in-denmark

https://doi.org/10.1016/j.dajour.2024.100486
Received 9 June 2023; Received in revised form 18 May 2024; Accepted 27 May 2024
Available online 28 May 2024
2772-6622/

city implementation. The vehicular network is a crucial component of the transportation system, which generates a massive amount of data that can be leveraged to improve traffic management and optimize routes. FL is a distributed machine learning technique that enables multiple edge devices to collaboratively train a model without sharing their raw data. In the context of smart cities, FL can be used to train machine learning models on data collected from vehicular networks while preserving the privacy and security of the data. The FL approach is particularly suitable for the vehicular network, where data is generated locally, and privacy and security concerns are associated with sharing the data.

This paper proposes a framework for FL in vehicular networks in smart cities. The proposed framework considers the challenges of limited network connectivity, privacy, and security concerns in such scenarios. The framework utilizes a hybrid approach that combines local training on individual vehicles with FL on a centralized server. The proposed framework is used to improve traffic management in smart cities, reduce congestion, and enhance the overall efficiency of the transportation system. Also in this real-life model, considering the total cost, distance, traffic and safety. Using genetic algorithm (GA) and FL makes this routing problem more effective and efficient for finding the optimum results.

The novelties of this investigation are as follows:

- A FL framework tailored for vehicular networks in smart cities. By addressing challenges like limited network connectivity, privacy, and security concerns, the framework offers a novel solution for collaborative model training.
- Show FL enhances sustainable routing in the Internet of Vehicular Things using genetic algorithm.
- Demonstrate FL enables collaborative, privacy-preserving, sustainable routing using vehicle intelligence for efficient model training.
- Derive safety, congestion, transportation, and fuel costs from the Internet of Things and machine learning.
- Obtain dynamic routing path for minimum travel cost or time smart/regular Internet of Things.
- An improved genetic algorithm with the integration of FL is proposed for obtained solutions. This integration enables the identification of optimal solutions while accommodating real-world constraints and dynamic environments.

The rest of the paper is organized as follows. Section 2 provides a detailed background on FL and vehicular networks in smart cities. Section 3 describes the proposed framework for FL in vehicular networks. Section 4 describes some proposed models. The proposed improved GA is explained in Section 5. The Section 6 presents the experimental evaluation of the proposed framework using a real-world dataset. Finally, Section 7 concludes the paper and highlights future research directions.

2. Literature review

2.1. Federated learning in smart city

Smart cities are designed to improve operational efficiency and provide a higher level of government service and citizen welfare by leveraging information and communication technology, improving inhabitants' quality of life and promoting economic growth [1]. The authors [2] proposed a local on-vehicle machine learning model updates are exchanged and verified in a distributed manner in an autonomous blockchain-based federated learning (BFL) system for privacy-aware and effective vehicular communication networking. According to [3], the most recent FL advances in industries such as the IoT, travel, communications, finance, and health. In a complete analysis of FL's existing and future advancements for smart cities [4]. We emphasize the socioeconomic, industrial, and technical developments driving FL for smart cities. The notion of FL for smart cities is then discussed, as well as various FL-integrated smart city applications, such as smart transportation systems, smart healthcare, smart grid, smart governance, smart disaster management, smart industries, and unmanned aerial vehicle (UAV) for smart city surveillance. FL reward models inherit the benefits of mobile crowd-sensing incentive systems, such as the capacity to assess a participant's utility more correctly. This is because evaluation approaches that may directly test the performance of their contributed model are more straightforward and concrete than evaluating the contribution of raw sensing data [5]. FL also enables users to acquire a higher reward from the same sensing data. As a result, a higher reward should be provided because there will be more significant computational overhead. As a result, consumers could earn higher compensation by supplying the same quantity of data at the expense of greater annoyance. This, along with the security benefits of FL, might motivate users to participate in FL-based sensing applications [6]. Vehicle-to-vehicle communication is critical for emerging automotive applications. As a result, when creating intelligent transportation systems, it is vital to incorporate ultra-reliable low-latency communication (URLLC) in-vehicle networks [7]. Using roadside units (RSU), vehicle users assess the tail dispersion locally [8]. The extremum theory characterizes the URLLC constraint, a tail distribution of the network scope queue length above a given threshold. Cloud-based learning approaches, on the other hand, are quite slow. The authors [9] proposes a context-aware ITS manages the services of intelligent vehicles. It handles the road traffic of traditional vehicles effectively, with a three-layered learning model that accounts for on-vehicle, on-co-vehicle, and on-fog-and-vehicle learning using a platoon control algorithm and FL at the Fog level. One recent study [10], reviews different privacy-preserving techniques used in FL, such as differential privacy, homomorphic encryption, and secure multi-party computation. The authors compare these techniques' privacy guarantees, communication and computation costs, and applicability to different FL scenarios. They suggest differential privacy is a promising technique for preserving privacy in FL. A comprehensively surveys the latest developments in FL for ITS. It explores the challenges in ITS and outlines the motivations for adopting FL. The survey covers existing deployments of FL in various ITS scenarios, including object recognition, traffic management, and service provision, while discussing potential issues [11]. In smart cities, the roadside units (RSUs) that are used for model aggregation according to [12], they also proposed FL architecture for real-time traffic estimation. Again, the FedSem method, which utilizes unlabeled data, is a semi-supervised FL technique for preserving privacy concerns [13]. A recent study on FL, used for energy forecasting in the power distribution system [14], The authors enhance the structure by leveraging adaptive learning, FL, and edge computing principles. It involves a central server aggregating multiple long short-term memory (LSTM) models.

2.2. Vehicular network in smart city

Today's increasing research on FL as a tool for protecting privacy while still allowing for learning. The authors of Ref. [15] have thoroughly analyzed FL, including its applications and the security and privacy concerns that must be addressed when using this approach. Their evaluation covers all the necessary aspects to ensure the security and privacy of FL, making it a reliable tool for privacy-protecting learning. In the investigation [16], they use contract theory to provide an incentive mechanism with asymmetric information and a continuum of kinds for RSU-vehicle interaction. The suggested mechanism meets the requirements of incentive compatibility and individual rationality while maximizing the RSU's profit. According to [17], the author provides a vehicle network secure FL with efficient communication scheme. We upload the model's updated parameters with local differential privacy to secure the privacy of the local update. To choose vehicles with better picture quality dynamically, we specifically suggest a greedy method, which maintains the cost of the system overall to a minimum in FL [18]. A survey in [19] explores Vehicular Edge Computing (VEC), explaining its concepts and technologies. It offers an overview of existing VEC architectures, discussing and exemplifying them through layered designs and the underlying vehicular communication supporting resource allocation mechanisms. According to [20] seeks to identify the best routes that meet all criteria by examining the waste collection procedure in Istanbul's Umraniye area. The Umraniye municipality gave information on the currently used routes, daily tonnages, number of truck journeys, and the quantity and position of containers in the Umraniye district. The recent study [21] talked about how edge computing architectures could be used in various offloading schemes for future vehicular computing applications. To reduce the infotainment content retrieval delay for smart automobiles, the authors of [22] formulated the joint optimization problem comprising communication, cashing, and computing capacities. A novel approach explores the concept of the green Internet of Vehicles (IoV) by examining considerations across five key scenarios: communication, computation, traffic, electric vehicles (EVs), and energy harvesting management [23]. Each scenario is analyzed regarding energy optimization, encompassing resource allocation, workload scheduling, routing design, traffic control, charging management, energy harvesting, and sharing. The literature relevant to these scenarios is compared, considering factors affecting energy efficiency such as resource limitations, channel state, network topology, and traffic conditions.

Most research assumes that all data owners provide such information to the task publisher truthfully [5]. According to [24] employed, the VCG mechanism entices selfish agents to play honestly by providing adequate rewards. Further studies have recently been conducted on FL as a privacy-preserving teaching tool. Regarding FL's security, the authors of [25] offered a thorough analysis of FL and its applications and confidentiality. They also demonstrated how to resolve this non-convex optimization. A recent study [26], used FL for classification while dealing with the imbalance and variable noise. Second, all of the data owners' private information, or data quality, must be known if the central system is to assign the FL task to them in the best way possible. Most studies suggest that all data owners will accurately disclose this information to the task publisher [5]. Distributed federated learning [27] proposes a heterogeneous neural network-based intrusion detection method for smart cities. It employs the most effective intruder detection technique, saving time and resources.

2.3. Meta-heuristics for routing in smart city

A very recent study [28,29] demand that GA works better than cutting-edge solution techniques since it finds very high-quality solutions quickly in job shop scheduling problems. The authors in [30] suggests a resource-sharing plan based on customer clustering to create an even allocation of spatial resources, hence lowering operational costs in a cooperative multi-depot logistics network, three-dimensional loading constraint and time window extended the evolutionary method for the multi-depot vehicle routing problem. Due to the computational difficulty involved in using the model to resolve real-world problems, a heuristic based on GA is also advised [31]. Utilizing the suggested method, sensitivity analysis is done and numerical situations are solved. In a recent study [32] integrates three domains: mobile edge computing, vehicular ad-hoc networks, and social networks to propose a new architecture called Social Vehicular Edge Computing (SoVEC). According to [33] considered customer happiness as the influencing element and the optimal cost as the goal function. The experiment compares the suggested adaptive GA and conventional GA to demonstrate the efficiency of the proposed algorithm. A recent study [34], offers a novel crew scheduling algorithm that enhances conventional crew scheduling by combining CPP and CRP into a single issue, and a new heuristic solution based on parallel GA. A worldwide heuristic search and adjustments for flights and personnel are part of the creative scheduling technique that enables crew scheduling. The suggested algorithm in [35] was tested on two significant road networks in Jordan, and its effectiveness was compared to that of a conventional GA. A study in [36] devised a framework effectively revealed design variable recipes and instructions for getting preferred structures with high permeability and filtration effectiveness throughout the actual production process. The authors according to [37] tackles the classical weighted multi-objective IoT service placement problem, with a focus on optimizing three parameters. The optimize three parameters are makespan, cost, and energy. Due to the non-convex nature of the solution space, the focus is on population-based meta-heuristic algorithms. This present investigation resolved the issue using an improved genetic algorithm (iGA) that included RW selection, cyclic crossover, and even mutation (generation dependent). The mutation operator has been improved, and this time, generation-based maintenance keeps the mutation operator more realistic. In the proposed iGA a generation-dependent even mutation is presented for the first time.

2.4. Intelligent routing services in smart city

The widespread usage of electric vehicles (EV) in the future is unavoidable. That will create a massive energy demand. As a result, keeping effective energy demand forecasting services for charging station (CS) suppliers is a pressing issue. Toll collectors and car manufacturers cannot share data due to privacy concerns. The investigator [38] built a model using encrypted entity alignment, secure FL, and prediction using data features on both sides and cross features between the two. A cross-feature model was also introduced, and the area of the curve (AUC) was enhanced. Ultimately, relatively centralized learning produced nearly lossless results. In the given study [39] authors proposed a CS-based decentralized, federated energy learning (DFEL) system for learning local datasets through CSs to estimate energy requirements properly and significantly cut communication costs. At a glance, we can observe the literature through Table 1 for better understanding.

Intelligent routing services in smart cities under IoT-enabled FL and vehicular networks can significantly improve traffic management and reduce congestion. With the rapid development of IoT technology and the emergence of FL, the collection and analysis of traffic data can be more efficient and secure than ever before. figure 1 shows the all scenario of a FL based vehicular network in smart city. Intelligent routing services involve machine learning algorithms to predict traffic patterns and optimize routes based on real-time data. This process can be enhanced by using FL to train machine learning models on the data generated by individual vehicles. By training the models on local data, the privacy and security of the data are preserved, while the accuracy of the models is improved. IoT sensors in vehicular networks can also provide real-time data about traffic patterns, road conditions, and weather conditions. This data can be combined with the data generated by individual vehicles to improve the accuracy of the machine-learning models. To overcome these challenges, sophisticated decision-making resource management based on IoT is frequently deployed. Analyze resource provisioning methods [51] and identify the variables that must be considered for optimal resource utilization in distributed systems.

Several research questions arise from this literature review.

I. How can FL be applied to vehicular networks in smart cities to improve traffic management and reduce congestion while preserving the privacy and security of the data?
II. What are the best machine learning algorithms and models that can be used in FL for vehicular networks in smart cities?
III. How can integrating IoT sensors in vehicular networks improve the accuracy and efficiency of FL models for traffic management and routing services?
IV. What are the technical challenges and limitations of implementing FL in vehicular networks, and how can they be addressed?
V. How can FL enable decentralized decision-making in smart city traffic management and routing services?

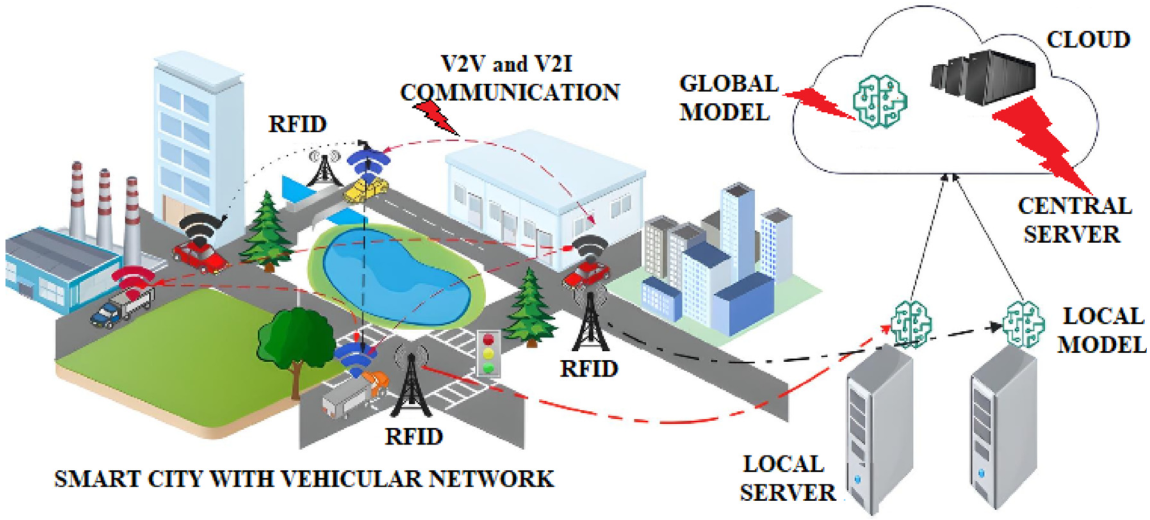


Fig. 1. Federated learning based vehicular network in smart city.

Table 1
Literature survey on FL in smart environment.

Scientific contributions	Analyzed features						
	IoVT	Machine learning	Edge computing	Fog computing	Cloud computing	Federated learning	Privacy preserving
Ayaz, Ferheen, et al. [40]	✓	✓	✓			✓	✓
Rizwan, Atif, et al. [41]	✓	✓			✓	✓	✓
Elbir, Ahmet M., et al. [42]	✓	✓			✓	✓	✓
Posner, Jason, et al. [43]		✓	✓	✓	✓		
Du, Zhaoyang, et al. [44]	✓	✓	✓			✓	✓
Fu, Lei, et al. [45]	✓	✓			✓	✓	✓
Tan, Kang, et al. [46]		✓			✓	✓	✓
Sathish, Routhu, et al. [47]	✓	✓	✓		✓		
Montazeri, Mina, et al. [16]	✓	✓	✓		✓	✓	
Dey et al. [48]		✓	✓	✓	✓		
Adelantado, Ferran, et al. [49]	✓	✓	✓		✓		
Apat et al. [37]		✓	✓	✓	✓		✓
Wazwaz, Amin, et al. [50]	✓	✓	✓	✓	✓		✓
Baidya, Sabur, et al. [21]	✓		✓	✓	✓		

VI. What are the implications of FL for the design of smart cities and transportation infrastructure?
VII. What are the ethical and legal considerations associated with using FL in vehicular networks, and how can they be addressed to ensure fair and equitable access to transportation services for all citizens?
VIII. How can FL be applied to environmental monitoring in smart cities using vehicular networks, and what are the potential benefits and challenges associated with this approach?
IX. What are the trade-offs between centralized and decentralized approaches to traffic management and routing services in smart cities, and how can FL help to reconcile these approaches?
X. How can FL enable adaptive and responsive traffic management systems that can adjust to changing conditions in real-time?

3. IoT-enabled information

3.1. Sensing data from vehicular networks

Let the set of vehicles in the network be denoted by $V = V_1, V_2, \dots, V_N$. Each vehicle V_i has a set of sensors that capture data about its environment, denoted by $S_i = s_1, s_2, \dots, s_m$. The sensors include GPS, accelerometer, gyroscope, camera, lidar, and radar. The goal is to collect and process the sensor data from all vehicles in the network to extract information about traffic patterns, road conditions, and other relevant information. Let $X = x_1, x_2, \dots, x_n$ represent the set of data points collected from all vehicles in the network, where each data point x_i is a feature vector of size d that captures relevant information about the vehicle's environment. Processing sensor data in vehicular networks uses clustering algorithms to group similar data points together. $C = C_1, C_2, \dots, C_K$. denote the set of clusters, where each cluster C_i is a subset of X that contains similar data points. A k-means clustering algorithm is used in vehicular networks. It partitions the data points into K clusters by minimizing the sum of squared distances between the data points and their respective cluster centroids. Let μ_j be the centroid of the jth cluster, given by:

$$\mu_j = \frac{\sum_j x_i}{N_j} \tag{1}$$

where N_j is the number of data points in the jth cluster.

The k-means algorithm is iterative applied to the data points until convergence is achieved when the clusters and their corresponding centroids are obtained. The centroids can then be used as representatives of the clusters to extract information about traffic patterns and road conditions. For example, the centroid of a cluster represents the average traffic flow rate or the average speed of vehicles in a particular area. This information is used to detect traffic congestion or accidents and to reroute vehicles to avoid congested or unsafe areas. Formulation of sensing data from devices in vehicular networks involves collecting and clustering data points from all vehicles to extract relevant information about traffic patterns and road conditions.

3.2. Proposed model for FL based vehicular networks for transportation

Let the set of vehicles in the network be denoted by $V = V_1, V_2, \dots, V_N$. Each vehicle V_i has a dataset of size m_i, denoted by $D_i = (x_i)|x_i \in$

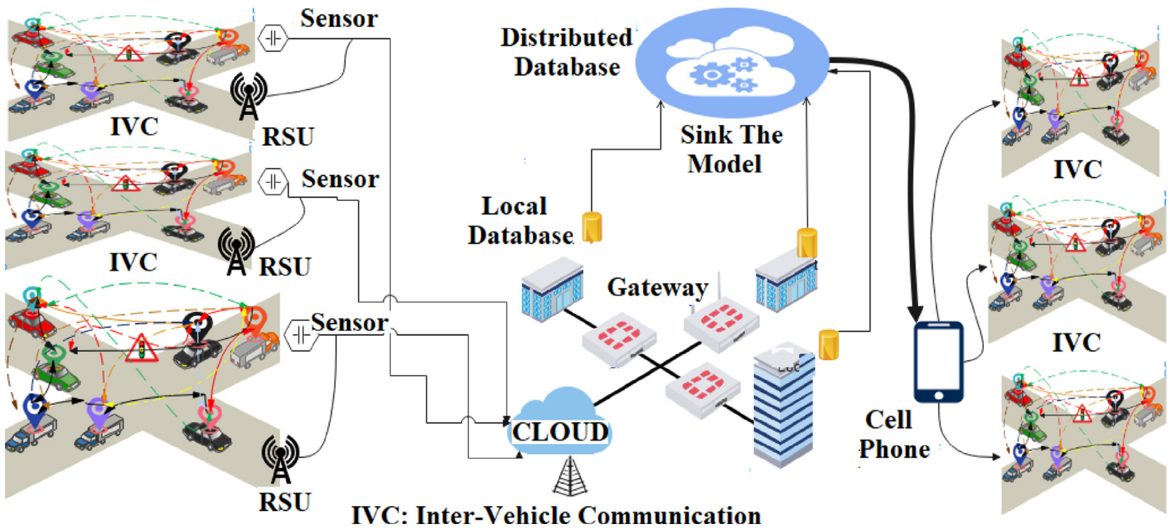


Fig. 2. Global Aggregation of Federated Learning in Vehicular Network.

$R_n, y_i \in R$. Each x_i represents the feature vector of the ith data point, and y_i represents its corresponding label. The goal is to train a global model θ^* on the entire dataset $D = \cup D_i, i = 1, 2, \ldots, N$, that minimizes a loss function $L(\theta)$, where θ represents the model parameters. Let $f_i(\theta)$ denote the local loss function of the ith vehicle, given by: $f_i(\theta) = \frac{1}{m_i} \sum_i (x_i, y_i) \in D_i$. L($\theta$; x_i, y_i) is the local model update for the ith vehicle is given by:

$\theta_{i,t+1} = \theta_{i,t} - \alpha \nabla f_i(\theta_{i,t})$, where α is the learning rate, and t represents the current iteration.

To perform FL, each vehicle V_i trains its model locally using its own dataset D_i and shares its updated model parameters with a central server. The central server aggregates the model updates from all vehicles and computes the updated global model parameters as follows: $\theta_{t+1} = \sum_{i=1}^{N} (w_i \theta_{i,t+1})$, where w_i represents the weight assigned to the ith vehicle based on its dataset size m_i. The weight is proportional to the size of the dataset, or it can be determined based on other factors, such as the reliability of the vehicle's network connection. The updated global model parameters θ_{t+1} are then sent back to each vehicle, which uses them to improve its local model in the next iteration. The FL process continues for a predetermined number of iterations or until convergence, at which point the global model θ^* is obtained. This model predicts the different objectives of traffic patterns and road conditions, improving driving experience and safety.

Acknowledging the challenges of limited network connectivity in vehicular networks is essential, yet these challenges still need comprehensive attention. This issue is particularly critical in the dynamic environment of vehicular networks, where vehicles are constantly moving and transitioning between areas with varying levels of network coverage. Limited network connectivity can result in communication delays, packet loss, and even temporary disconnections, impacting crucial tasks such as real-time data sharing, vehicle-to-vehicle communication, and updates to FL models. Failure to thoroughly address these challenges can lead to inefficient traffic management, compromised safety systems, and degraded overall network performance. This may involve adaptive communication protocols, intelligent routing algorithms, and innovative network infrastructure designs tailored to the unique requirements of vehicular environments. Only through comprehensive and proactive measures can the challenges of limited network connectivity be effectively mitigated in the dynamic landscape of vehicular networks.

3.3. Vehicular edge computing

Vehicular edge computing (VEC) is an emerging computing paradigm that combines the capabilities of edge computing and vehicular networks to support the growing demand for real-time applications and services in vehicular environments. It involves deploying edge computing nodes near vehicular networks to enable computation, storage, and communication capabilities. VEC can be used to support a variety of applications, including safety-critical applications, intelligent transportation systems, and entertainment applications.

The main objective of VEC is to provide low-latency, high-bandwidth, and reliable communication and computation services to vehicles, passengers, and other users in the vehicular environment. Fig. 2 shows the global aggregation for collecting data from different vehicular users. This is achieved by deploying edge computing nodes near the vehicles, such as small cell base stations, access points, and servers. The edge computing nodes are on the road infrastructure, such as traffic lights, lamp posts, or vehicles. VEC provides several benefits compared to traditional cloud-based computing approaches. One of the main advantages is the reduction of latency and bandwidth requirements for communication between vehicles and the cloud. This is because edge computing nodes are near the vehicles, allowing faster and more reliable communication. Another benefit is the improved reliability and availability of services, as VEC can continue to operate even when the cloud is unavailable or has limited connectivity. In intelligent transportation applications, VEC supports entertainment applications such as in-vehicle communication systems, RSU, V2X, etc. These applications require high-bandwidth communication and computation capabilities to provide a seamless user experience.

Overcoming the latency and synchronization challenges inherent in straddling two types of training, such as local vehicle training and centralized server of FL, necessitates combining strategic approaches. Optimized communication protocols are paramount to minimizing latency and ensuring efficient data transmission between vehicles and the central server (see Fig. 3). Employing asynchronous updates allows for decoupling local training from server interactions, mitigating the need for real-time synchronization and facilitating smoother integration of updates. Predictive synchronization, achieved through proactive algorithms, anticipates local training updates' availability and schedules synchronization accordingly, preemptively managing data transfers to reduce delays. Leveraging edge computing capabilities enables performing specific computation and training tasks directly on vehicles or at the network edge, alleviating the need for frequent data transmission to the central server. Dynamic resource allocation ensures sufficient resources for local training and server interactions, adapting to current workload and network conditions to prevent bottlenecks. Incorporating federated averaging algorithms that account for time delays between local updates and server synchronization allows for efficient aggregation despite temporal disparities. Robust error-handling mechanisms and

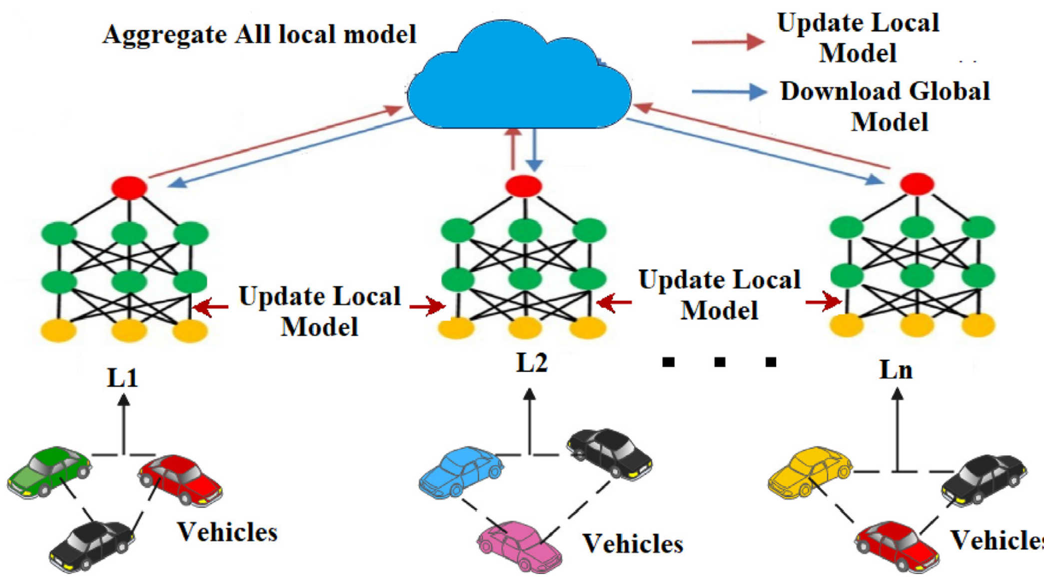


Fig. 3. Federated learning based local and global aggregation.

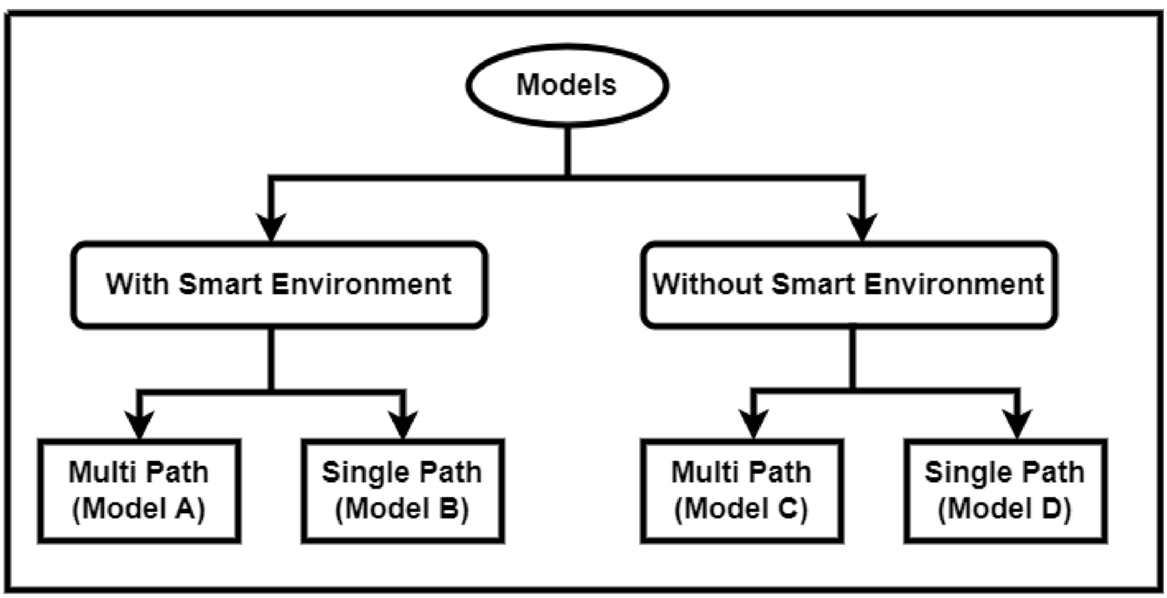


Fig. 4. Different models of proposed system.

resilience measures address communication failures and disruptions, ensuring system stability and continuity. The strategic implementation of these approaches can effectively mitigate the challenges associated with straddling diverse training types, fostering efficient coordination, and optimizing overall performance.

4. Proposed models in smart city

Here, we are implementing a new model combining all these based on the above models. Consider energy management, traffic prediction, public safety, smart parking, and optimal routing for the proposed model. All proposed models are presented in Fig. 4. Also, consider multipath and single-path models with and without a smart environment.

$$Minimize\ Z = \underbrace{\frac{\sum_{i=1}^{N-1} c(x_i,x_{i+1},r_{ii+1})*dis(x_i,x_{i+1},r_{ii+1})e(x_N,x_1,r_{N1})}{Fuel\ cost}} + \frac{\sum_{i=1}^{N-1} t(x_i,x_{i+1},r_{ii+1})*t(x_N,x_1,r_{N1})}{Traffic} + \frac{\sum_{i=1}^{N-1} s(x_i,x_{i+1},r_{ii+1})*t(x_N,x_1,r_{N-1})}{Safety} + \frac{\sum_{i=2}^{N} p_i*d_i}{Parking} \tag{2}$$

In Eq. (2), four parts are included. The first portion of the equation denotes the fuel expense and travel costs of the vehicle on many paths, where $c(x_i, x_{i+1}, r_{ii+1})$ is the journey expense per unit of distance. The second section displays the expense of traffic along the entire route. The third path is used for the safety measurement. Parking fees are required at every node and are shown in the last section of the calculation.

The urban areas are characterized by dense populations and high vehicular traffic, posing unique challenges for scalability in vehicular networks. As the number of connected vehicles increases, the system must efficiently manage and process a growing volume of data while maintaining performance and reliability. Factors such as network congestion, resource allocation, and communication overhead become increasingly significant and require careful consideration. Additionally, scalability issues may arise concerning infrastructure deployment, protocol efficiency, and algorithm optimization. By thoroughly exploring these challenges and proposing scalable solutions tailored to urban environments, the paper can provide valuable insights into the feasibility and effectiveness of implementing federated learning in such dynamic and densely populated settings.

Integrating the system into established smart cities without requiring extensive infrastructure changes is a valid concern. Smart cities often have pre-existing infrastructure and systems, which may need help accommodating new technologies or implementations. Retrofitting existing infrastructure to support the system's requirements could be costly, time-consuming, and disruptive to urban operations. To address this concern, it is essential to develop the system focusing on scalability, flexibility, and compatibility with existing smart city infrastructure. Leveraging open standards, modular architectures, and interoperable protocols can facilitate smoother integration while minimizing the need for extensive infrastructure changes. Collaboration between stakeholders, including city authorities, technology providers, and infrastructure operators, is also crucial to ensure successful integration and alignment with the broader goals of smart city initiatives. Now different components are discussed in coming subsections.

4.1. Energy management in a smart city through FL

Energy management in a smart city through FL involves using machine learning algorithms to optimize energy consumption across various devices and systems in the city while preserving data privacy. FL allows multiple devices and systems to collaborate and share their data without sharing it.

Algorithm 1: Federated Learning for Energy Management in Smart City through Vehicular Network

Input : Training data (D_i) for each vehicle (V_i) in the vehicular network, total number of iterations (E), learning rate (η), communication rounds (C), local epochs (E), and energy threshold (ϵ).

Output: Global model M for energy management in smart city.

Initialization: $M \leftarrow$ Randomly initialized model
for $t = 1$ **to** E **do**
    $S_t \leftarrow$ Random subset of vehicles in the network
    for $c = 1$ **to** C **do**
        for $V_i \in S_t$ **do**
            $M_i \leftarrow$ Copy of M
            Train M_i on D_i for E epochs using SGD with learning rate η
            if *Energy level of* $V_i < \epsilon$ **then**
                Skip communication with V_i
            end
            else
                Send M_i to all other vehicles in S_t
                Receive models from other vehicles and average them to update M_i
            end
        end
    end
    $M \leftarrow$ Average of all M_i
end

$$
\begin{aligned}
\text{minimize:} \quad & \sum_{t=1}^{T} c_t(\alpha_t - \beta_t) \\
\text{subject to:} \quad & \sum_{i=1}^{N} p_{i,t} \le P_{\max,t}, \quad \forall t = 1, \dots, T \\
& \sum_{t=1}^{T} p_{i,t} \le E_{i,\max}, \quad \forall i = 1, \dots, N \\
& E_{i,t} = E_{i,t-1} - \eta_i p_{i,t}, \quad \forall i = 1, \dots, N, t = 2, \dots, T \\
& E_{i,1} = E_{i,\text{init}}, \quad \forall i = 1, \dots, N \\
& \alpha_t - \beta_t = \sum_{i=1}^{N} p_{i,t} - D_t, \quad \forall t = 1, \dots, T \\
& \alpha_t, \beta_t \ge 0, \quad \forall t = 1, \dots, T \\
& p_{i,t} \ge 0, \quad \forall i = 1, \dots, N, t = 1, \dots, T,
\end{aligned}
$$

where T is the total number of time periods, N is the total number of energy-consuming devices, $p_{i,t}$ is the power consumption of device i at time period t, $P_{\max,t}$ is the maximum power capacity at time period t, $E_{i,\max}$ is the maximum energy capacity of device i, $E_{i,t}$ is the energy level of device i at time period t, η_i is the efficiency of device i, D_t is the energy demand at time period t, and α_t and β_t are auxiliary variables.

The objective is to minimize the difference between the total energy consumption and the energy demand at each time period, weighted by the cost c_t of energy at that time. The first constraint ensures that the power consumption of all devices at each time period does not exceed the maximum capacity. The second constraint ensures that the energy consumption of each device over all time periods does not exceed its maximum capacity. The third constraint tracks the energy level of each device over time, taking into account the efficiency η_i of each device. The fourth constraint sets the initial energy level of each device to a given value $E_{i,\text{init}}$. The fifth constraint defines the auxiliary variables α_t and β_t, which are used to calculate the difference between the total energy consumption and demand at each time period. The last three constraints ensure that all variables are non-negative.

This formulation is used to develop optimization algorithms (see Algorithm 1) for energy management in a smart city, for example, by incorporating additional constraints such as renewable energy sources or energy storage systems.

4.2. Federated learning in traffic prediction

Traffic prediction in a smart city through FL involves using machine learning algorithms to predict traffic patterns across various locations in the city while preserving data privacy. FL allows multiple devices and systems to collaborate and share their data without sharing it.

By using FL in traffic prediction, smart cities optimize traffic flow, reduce congestion, and improve road safety while preserving data privacy. This led to significant cost savings, improved air quality, and enhanced quality of life for residents and visitors in the city.

Introducing a centralized server for FL to enhance privacy comes with its own security vulnerabilities. While centralization offers benefits such as data localization and aggregate learning, it concentrates sensitive information in one location, making it a prime target for malicious attacks. A centralized server becomes a single point of failure, susceptible to breaches that could result in unauthorized access to vast amounts of data or manipulation of the learning process. Even with privacy-enhancing measures like differential privacy, the risk of privacy breaches remains if the centralized server is compromised. To address these vulnerabilities, robust security measures such as encryption, access control, anomaly detection, and regular model verification are essential. A prominent emphasis on local training risks introducing region-specific biases, potentially modifying the model's broader relevancy. Training models predominantly on data from a specific region may lead to the model learning patterns, trends, and characteristics unique to that area, resulting in biases that may not generalize well to other regions or diverse populations.

Here variables are defined as:

θ : model parameters

$x_i(t)$: traffic volume at node i at time t

$y_{ij}(t)$: traffic volume on edge (i, j) at time t

Algorithm 2: Federated Learning for Traffic Prediction in Smart City through Vehicular Network

Input : Training data D_i for each vehicle V_i in the vehicular network, total number of iterations T, learning rate η, communication rounds C, local epochs E, and prediction horizon H.
Output: Global model M for traffic prediction in smart city.

Initialization: $M \leftarrow$ Randomly initialized model
for $t = 1$ **to** T **do**
  $S_t \leftarrow$ Random subset of vehicles in the network
  for $c = 1$ **to** C **do**
    for $V_i \in S_t$ **do**
      $M_i \leftarrow$ Copy of M
      Train M_i on D_i for E epochs using SGD with learning rate η
      Send M_i to all other vehicles in S_t
      Receive models from other vehicles and average them to update M_i
    end
  end
  $X_t \leftarrow$ Data for prediction horizon H
  $y_t \leftarrow$ True values for prediction horizon H
  Predict $\hat{y}_t$ using M
  $M \leftarrow$ Average of all M_i
end

The proposed objective:

Minimize the prediction error of traffic volume on all nodes and edges over a time horizon of T, using a FL approach:

$$\min_{\theta} \quad \sum_{i=1}^{N} w_i \sum_{t=1}^{T} (\hat{x}_i\theta(t) - x_i(t))^2 + \sum_{i=1}^{N} w_i \sum_{j\in N_i} \sum_{t=1}^{T} (\hat{y}_{ij}\theta(t) - y_{ij}(t))^2$$
$$\text{s.t.} \quad \theta_i = \theta, \quad \forall i \in \mathcal{I}$$

where:

N = set of nodes in the city
N_i = set of neighbors of node i
w_i = weight assigned to node i
$\hat{x}_i\theta(t)$ = predicted traffic volume at node i at time t using the model parameters θ
$\hat{y}_{ij}\theta(t)$ = predicted traffic volume on edge (i, j) at time t using the model parameters θ
$\mathcal{I}$ = set of devices participating in the FL process

The objective function minimizes the squared difference between predicted and actual traffic volumes on all nodes and edges over a time horizon of T, using a FL approach (see Algorithm 2) where each device (i.e., node) computes a local gradient and sends it to a central server for aggregation. The constraints ensure that all devices use the same model parameters.

4.3. Public safety in a smart city through federated learning

Public safety in a smart city through FL involves using machine learning algorithms to analyze data from various sources to identify potential safety threats and prevent them while preserving data privacy. FL allows multiple devices and systems to collaborate and share their data without sharing it. In public safety, smart cities improve emergency response times, reduce crime, and enhance overall safety while preserving data privacy using the FL. This increased public trust and a sense of security among residents and visitors to the city.

Here variables are defined as:

x_i : binary variable indicating whether node i is a safe location
y_{ij} : binary variable indicating whether edge (i, j) is a safe route

The proposed objective: Maximize the overall public safety in the city, which can be expressed as the sum of the safety scores of all nodes and edges:

$$\begin{aligned} \max \quad & \sum_i s_i x_i + \sum_{i,j} s_{ij} y_{ij} \\ \text{s.t.} \quad & x_i \le z_i, \quad \forall i \\ & y_{ij} \le x_i, \quad \forall i, j \in E \\ & y_{ij} \le x_j, \quad \forall i, j \in E \\ & x_i + x_j - y_{ij} \le 1, \quad \forall i, j \in E \end{aligned}$$

where:

E = set of edges in the city
z_i = binary variable indicating whether node i is available for improvement
s_i = safety score of node i
s_{ij} = safety score of edge (i, j)

The constraints ensure that a node is only safe if it is available for improvement, an edge only be safe if both endpoints are safe, and a path only be safe if all of its edges are safe.

4.4. FL-based smart parking in a smart city

Smart parking in a smart city through FL involves using machine learning algorithms to optimize the use of parking spaces across various locations in the city while preserving data privacy. FL allows multiple devices and systems to collaborate and share their data without sharing it.

By using FL in smart parking, smart cities optimize parking spaces, reduce traffic congestion, and improve the overall parking experience for drivers while preserving data privacy. This lead to increased revenue for parking providers, reduced carbon emissions, and enhanced quality of life for residents and visitors in the city.

$$\begin{aligned} \text{minimize:} \quad & \sum_{i=1}^{N} c_i x_i \\ \text{subject to:} \quad & \sum_{i=1}^{N} x_i \le C, \\ & x_i \in \{0, 1\}, \quad \forall i = 1, 2, \dots, N \end{aligned}$$

where N is the total number of parking spots, x_i is a binary variable that indicates whether spot i is occupied (1) or not (0), C is the total number of available parking spots, and c_i is the cost associated with parking in spot i.

The objective is to minimize the total cost of parking (see Algorithm 3), which is the sum of the cost of parking in each occupied spot. The constraint ensures that the total number of occupied spots does not exceed the total number of available spots. The decision variable x_i takes a value of 1 if spot i is occupied and 0 otherwise, which ensures that each spot is occupied by only one vehicle at a time.

4.5. Formulation of optimal route planning in smart city through FL

Optimal route planning in a smart city through FL involves using machine learning algorithms to find the best routes for vehicles to navigate the city when traversing from one location to another location. FL allows multiple devices and systems to collaborate and share their data without sharing it. Using FL in optimal route planning, smart

Algorithm 3: Federated Learning for Smart Parking in Smart City through Vehicular Network

Input : Training data (D_i) for each vehicle (V_i) in the vehicular network, total number of iterations (P), learning rate (η), communication rounds (C), local epochs (E), and occupancy threshold (θ).
Output: Global model M for smart parking in smart city.

```
Initialization: M ← Randomly initialized model
for t = 1 to P do
    S_t ← Random subset of vehicles in the network
    for c = 1 to C do
        for V_i ∈ S_t do
            M_i ← Copy of M
            Train M_i on D_i for E epochs using SGD with
              learning rate η
            Send M_i to all other vehicles in S_t
            Receive models from other vehicles and average
              them to update M_i
        end
    end
    X_t ← Data for current occupancy
    y_t ← True occupancy
    Predict ŷ_t using M
    if ŷ_t > θ then
        Find nearby parking spaces using smart parking
          algorithm
        Send parking space information to nearby vehicles
    end
    M ← Average of all M_i
end
```

Algorithm 4: Federated Learning for Optimal Route Planning in Smart City through Vehicular Network

Input : Training data (D_i) for each vehicle (V_i) in the vehicular network, total number of iterations (R), learning rate (η), communication rounds (C), local epochs (E), starting point (s), destination (d), and traffic congestion level (T_c).
Output: Global model M for optimal route planning in smart city.

```
Initialization: M ← Randomly initialized model
for t = 1 to R do
    S_t ← Random subset of vehicles in the network
    for c = 1 to C do
        for V_i ∈ S_t do
            M_i ← Copy of M
            Train M_i on D_i for E epochs using SGD with
              learning rate η
            Send M_i to all other vehicles in S_t
            Receive models from other vehicles and average
              them to update M_i
        end
    end
    X_t ← Data for current traffic congestion level T_c
    Predict ŷ_t, the optimal route from s to d, using M
    Send ŷ_t to nearby vehicles
    M ← Average of all M_i
end
```

cities reduce travel time, improve fuel efficiency, and decrease traffic congestion while preserving data privacy. This lead to significant cost savings, reduced carbon emissions, and enhanced quality of life for residents and visitors in the city (see Algorithm 4).

$Variables$:

x_{ij} : binary variable indicating whether arc (i, j) is selected in the solution or not

u_i : continuous variable representing the cumulative demand up to node i

f_{ij} : continuous variable representing the flow of vehicles from node i to node j

Objective: Minimize the total cost, which is expressed as the sum of the distances or travel times between all nodes in the solution, plus penalties for congestion:

$$
\begin{aligned}
\min \quad & \sum_{i} \sum_{j \in D} (c_{ij} + p_{ij} f_{ij} + e_{ij} x_{ij} + l_{ij} x_{ij}) \\
\text{s.t.} \quad & \sum_{j \in D} x_{ij} = 1, \quad \forall i \in C \\
& \sum_{i \in C} x_{0i} = \sum_{i \in C} x_{i0} = m \\
& u_i + q_j \leq u_j, \quad \forall (i, j) \in A \\
& u_i \geq q_i, \quad \forall i \in C \\
& u_i - u_j + Q x_{ij} \leq Q - q_j, \quad \forall (i, j) \in A,\ i \neq 0, j \neq 0 \\
& \sum_{k \in C, j \in C} f_{kj} \leq C_{ij}, \quad \forall i \in C \\
& \sum_{i \in C, j \in C} e_{ij} x_{ij} \leq E, \\
& \sum_{i \in C, j \in C} l_{ij} x_{ij} \leq P.
\end{aligned}
$$

Where:

$D = C \cup 0, \quad C$ is the set of customers, and 0 is the depot

A = set of arcs

q_i = demand of customer i

Q = capacity of each vehicle

m = number of vehicles

c_{ij} = cost of traveling from node i to node j

p_{ij} = penalty for congestion on arc (i, j)

e_{ij} = emissions generated by traveling from node i to node j

l_{ij} = penalty for making a left turn from node i to node j

C_{ij} = capacity of the road segment between nodes i and j

4.6. Theoretical feasibility analysis of the framework

Assume the objective function of the genetic algorithm used for sustainable routing in the Internet of Vehicular Things (IoVT). Let us denote the objective function as $f(x)$, where x represents the chromosome or solution vector encoding vehicle routing decisions. By formulating $f(x)$ to incorporate criteria such as energy efficiency, traffic congestion, and environmental impact, we can quantitatively evaluate the quality of routing solutions produced by the genetic algorithm. Furthermore, we can analyze the computational complexity of the genetic algorithm, considering factors such as the population size (N), chromosome length (L), and number of generations (G). If we denote the time complexity of the genetic algorithm as $O(N * L * G)$, we can assess how the computational requirements scale with increasing problem size.

5. Proposed improve GA in FL with vehicular network

Since the proposed models are $NP-hard$, we used a metaheuristic termed as an improved genetic algorithm (iGA) to solve the problem with Roulette wheel (RW) selection, cyclic crossover, and even mutation (generation dependent). The following steps of a genetic algorithm are as follows in the given subsections:

5.1. Roulette wheel selection

In a genetic algorithm, the RW selection (Algorithm 5), also known as fitness-proportional selection or stochastic universal sampling, is used to select individuals for reproduction based on their fitness values. The approach is given in Algorithm 5.

The RW selection process gives individuals with higher fitness values a greater chance of being selected, but it still allows individuals with lower fitness values to have a chance of being chosen.

Algorithm 5: Roulette Wheel Selection

Input : Population P of N individuals, Fitness values F
Output: Selected individual from P

$S \leftarrow \sum_{i=1}^{N} F_i$; `// Calculate the sum of fitness values`
$r \leftarrow$ a random number between 0 and S ; `// Generate a random number`
$c \leftarrow 0$; `// Initialize the cumulative fitness`
for $i \leftarrow 1$ **to** N **do**
  $c \leftarrow c + F_i$; `// Calculate the cumulative fitness`
  if $c \geq r$ **then**
    return Individual i ; `// Select the corresponding individual`

5.2. Cyclic crossover

Cyclic crossover (Algorithm 6) is a genetic operator used in genetic algorithms to create offspring by exchanging genetic material between two parent individuals. This strategy preserves the order and distribution of genetic material in the parents and is used for permutation-based representations. It aims to create diverse offspring by exchanging genetic information while maintaining some of the parent's characteristics.

Algorithm 6: Cyclic Crossover

Input : Parent individuals P_1 and P_2 with chromosome length L
Output: Offspring individuals O_1 and O_2

$cycle \leftarrow$ a random cycle of length k in P_1 ; `// Select a random cycle`
$O_1 \leftarrow$ a copy of P_1 ; `// Initialize the first offspring`
$O_2 \leftarrow$ a copy of P_2 ; `// Initialize the second offspring`
for $i \leftarrow 1$ **to** L **do**
  if $i \in cycle$ **then**
    $O_1(i) \leftarrow P_2(i)$; `// Swap genes in offspring 1`
    $O_2(i) \leftarrow P_1(i)$; `// Swap genes in offspring 2`
return O_1, O_2 ; `// Return the two offspring`

5.3. Even mutation (generation dependent):

Instead of deterministic (predefined) p_m, in the present study, a dynamic adaptation of p_m is introduced. In the present study, p_m is evaluated depending on generation by a given function.

(a) Random mutation: Random mutation (Algorithm 7) is a genetic operator used in GA to introduce small, random changes in an individual's genetic material.

Random mutation is used to introduce exploration and prevent premature convergence in genetic algorithms. By randomly altering genes, it allows for the exploration of new regions in the search space, potentially leading to improved solutions. The mutation probability determines the likelihood of each gene being mutated.

Algorithm 7: Random Mutation

Input : Individual I with chromosome length L, Mutation probability p_m
Output: Mutated individual I'

$I' \leftarrow$ a copy of I ; `// Initialize the mutated individual`
for $i \leftarrow 1$ **to** L **do**
  $r \leftarrow$ a random number between 0 and 1 ; `// Generate a random number`
  if $r \leq p_m$ **then**
    $g \leftarrow$ a random gene value ; `// Generate a random gene`
    $I'(i) \leftarrow g$; `// Mutate the gene at position i`
return I' ; `// Return the mutated individual`

(b) Even mutation (generation dependent):
Here, we use generation-dependent even mutation as
$p_m = \frac{k}{\sqrt{even\ number\ corresponding\ current\ generation\ number}}$, k∈[0,1].
For even numbers 2, 4, 6, 8, ⋯, p_m will be $\frac{k}{\sqrt{2}}, \frac{k}{\sqrt{4}}, \frac{k}{\sqrt{6}}, \cdots$ as generation progress.

(c) Mutation process:
If $r < p_m$, $r \in [0,1]$ is satisfied, then the corresponding solution is selected for mutation. Finally, conventional random mutation is performed depending on the value of p_m.

6. Results and discussion

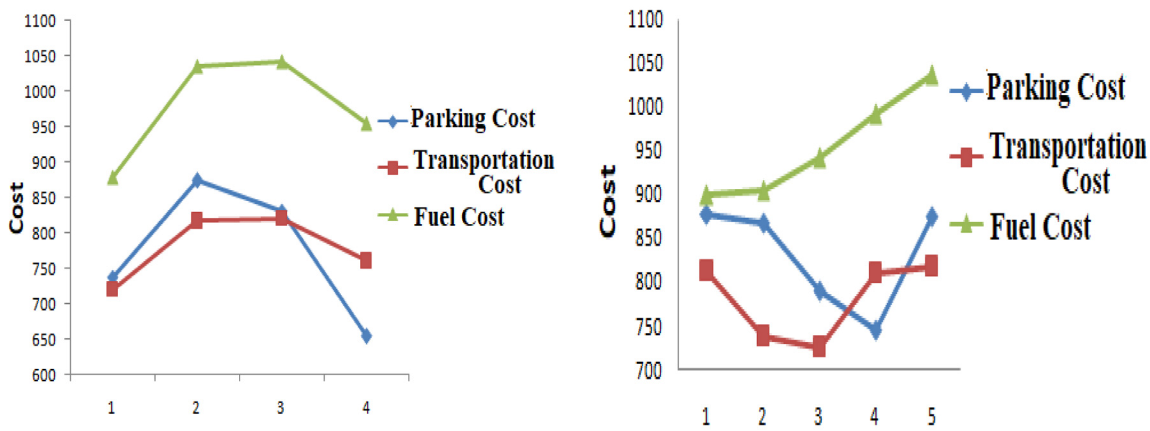
All the models are solved through two heuristic algorithms- iGA (proposed algorithm) and ACO [52]. Here, all costs are considered in rupees (Rs) and distance in kilometer (km). A comparison between scenarios with and without a smart environment is illustrated in the provided Fig. 5.

Fig. 5 represents how a smart environment impacts parking, transportation, and fuel costs a lot. In all cases, smart environment performs better than normal environment routing.

Based on the results, it is observed that for all the models, multipath performs better in smart environment (Fig. 5). Also, all the results are compared with another heuristic algorithm (ACO) (c.f Tables 2, 3, 4, 5, 6, 7 and Figs. 6(a), 6(b), 6(c), 6(d), 6(e)).

For Model A, from Table 2 and Fig. 6(a) it is shown that without considering smart conditions, iGA with multipath routing as 0(1)- 9(1)-7(2)-4(1)-2(0)-6(0)-8(1)-1(1)-5(1)-3(1), with parking cost (Rs 938.4) total distance covered (92.74 km), transportation cost (Rs 721.94), fuel cost (Rs 879.83) and finally total safety (5.38 unit). Similarly, for the single path, it increases costs as well as decreases the safety level too. The results for single paths, without considering a smart environment using iGA and ACO, are as follows: 0-8-6-7-4-3-2-5-1-9, parking cost (Rs 904.01), distance (100.68 km), transportation cost (Rs 831.66), fuel cost (931.64 km), obtained safety (3.3 unit). Therefore, the multipath approach consistently outperforms the single path approach in terms of total cost, distance covered, and safety.

For Model B, from Table 3 and Fig. 6(b) it is shown that considering smart conditions, ACO with multipath routing as 0(2)-4(1)-8(0)-6(1)-9(2)-3(0)-5(0)-1(2)-7(2)-2(0) with parking cost (Rs 874.45), total distance covered (102.8 km), transportation cost (Rs 817.25), fuel cost (Rs 1036.28), and accumulated safety (2.85 unit). On the other hand, the single path study using ACO yields the following results: 0-8-2-5-6-1-3-4-7-9, with a parking cost of Rs 818.8, a total distance of 99.2 km, a transportation cost of Rs 814.34, a fuel cost of Rs 1005.08, and an achieved safety level of 4.98 units.



(a) Smart Environment (b) Normal Environment

Fig. 5. Cost Comparison: smart vs. without smart environments.

For Model C, from Table 4 and Fig. 6(c) it is shown that without considering smart conditions, iGA with multipath routing as 0(1)-6(1)-2(0)-5(0)-1(1)-9(1)-7(0)-4(2)-8(0)-3(2) with parking cost (Rs 831.51), total distance (100.74 km), transportation cost (Rs 821.84), fuel cost (Rs 1043.26) and obtained safety (2.22 unit). The single path study shows the route as 0-4-7-1-8-6-9-2-5-3, with a parking cost of Rs 919.2, a total distance of 101.58 km, a transportation cost of Rs 841.34, a fuel cost of Rs 1026.72, and a safety level of 4.24 units.

For Model D, from Table 5 and Fig. 6(d) it is shown that considering smart conditions, ACO with multipath routing as 0(0)-8(1)-5(1)-7(2)-9(1)-4(0)-1(1)-3(2)-2(2)-6(1) with parking cost (Rs 654.89), distance covered (97.24 km), transportation cost (Rs 761.3), fuel cost (Rs 955.33), and safety (4.94 unit). Similar observations were found for the single-path model, with the route as 0-8-6-7-4-3-2-5-1-9. The corresponding metrics are as follows: parking cost Rs 904.01, distance covered 100.68 km, transportation cost Rs 831.66, fuel cost Rs 931.6, with an attained safety level of 3.3 units.

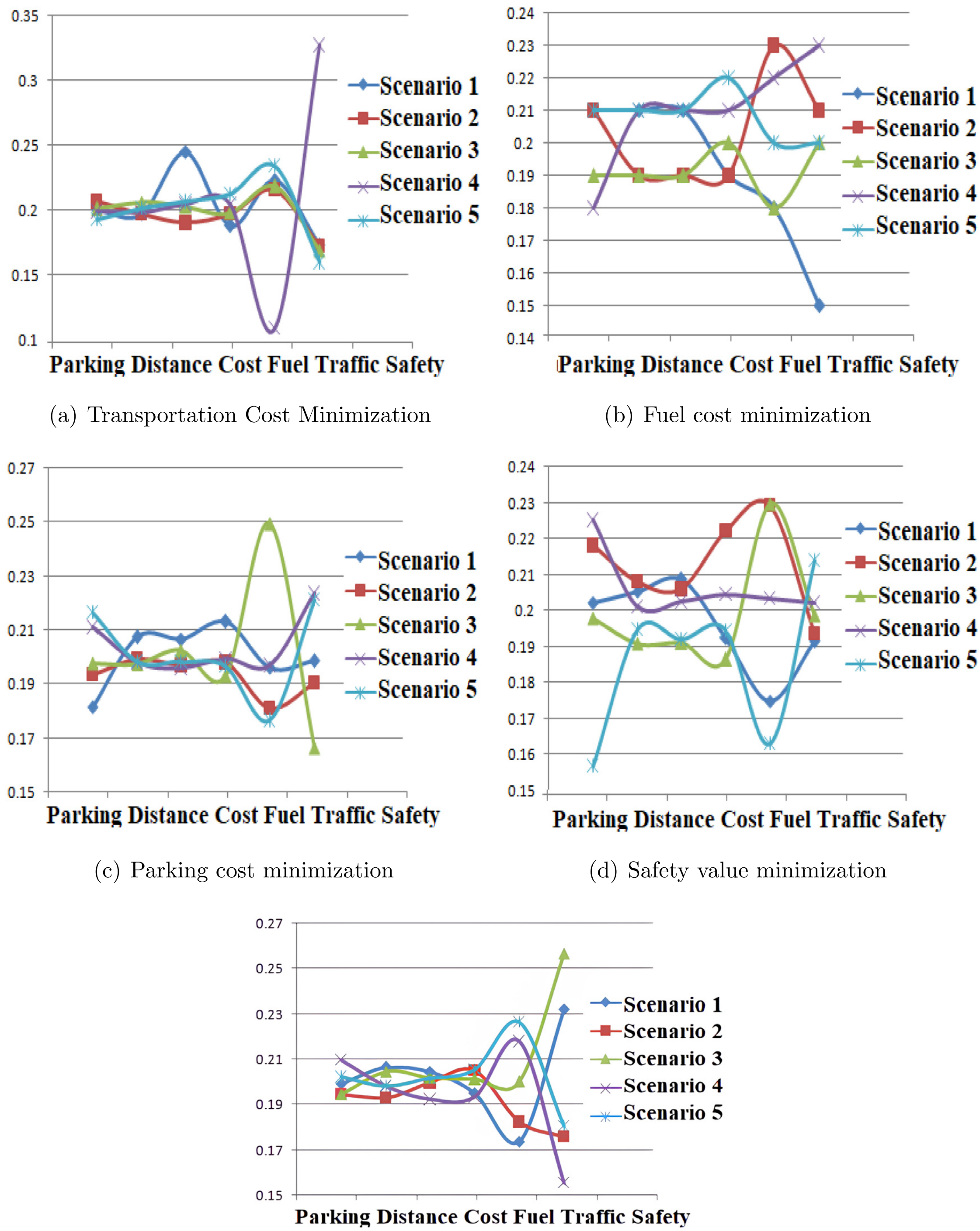
The analysis conducted for Models A, B, C, and D reveals insightful findings regarding the effectiveness of different routing strategies under various conditions. For instance, in Model A, without considering smart conditions, iGA with multipath routing demonstrated superior performance in terms of cost, distance covered, and safety compared to the single-path approach. Conversely, in Model B, with the inclusion of smart conditions, ACO with multipath routing exhibited favorable outcomes, highlighting the importance of incorporating intelligent elements in routing algorithms. Similarly, in Models C and D, the comparison between multipath and single-path routing strategies underscored the benefits of multipath routing in optimizing various performance metrics, including transportation cost, fuel cost, parking cost, and safety. Overall, the findings emphasize the significance of intelligent routing solutions in enhancing efficiency and effectiveness in smart environment. The Figs. 6(a) to 6(e) depicting the minimized transportation cost, fuel cost, parking cost, safety, and traffic provide a comprehensive overview of the optimization outcomes, enabling informed decision-making by management based on specific operational requirements and priorities.

Relying on a single real-world dataset to validate the framework raises concerns regarding its generalizability and applicability across diverse city landscapes which is consider for the present study. Urban environments vary significantly in traffic patterns, infrastructure layout, population density, and other factors influencing the performance and effectiveness of vehicular network frameworks. Using more than one dataset limits the scope of the validation process and may not capture the full spectrum of challenges and scenarios encountered in different cities. Researchers should consider incorporating multiple datasets with diverse characteristics from various urban areas to address this limitation. This approach enables a more comprehensive evaluation of the framework's robustness, scalability, and adaptability across different city landscapes. Additionally, simulations or experiments in controlled environments can complement real-world data validation, providing insights into the framework's performance under varying conditions. By adopting a more comprehensive validation approach, researchers can enhance confidence in the framework's broad applicability and reliability across diverse urban landscapes. But the present study is limited with a singular real-world dataset.

In Fig. 7, its shown that accuracy depends on the number of epochs. This is observed that the accuracy of the model is 90.63%.

7. Conclusions

With increasing interest in FL shown by businesses and academic institutions, it is essential to discuss applying FL in vehicular IoT environments. This study explores the application of FL in vehicular IoT environments, aiming to enhance traffic management and optimize routes in smart cities. Here, we propose an integration of FL on a centralized server with local training on individual cars, the proposed framework addresses challenges such as restricted network connectivity, privacy issues, and security concerns. Thus the studies shown in a systematic way by the work that have been done, the technical problems, possible solutions, and unanswered questions about using FL in-vehicle IoT. First, we studied the FL's most recent attempts and discussed its uses and problems in wireless IoT situations. In-depth discussions on the technical difficulties were conducted while reviewing the studies that have already been done on the use of FL for vehicular IoT. Subsequently, we introduced various models within the domain of intelligent transportation systems. To address these models, we devised an improved GA and compared it with the standard ACO algorithm. In most cases, the improved GA demonstrated superior performance. The next topic of discussion is the future paths for research on integrating FL with vehicular IoT, taking into account both FL's use of IoT and the development of IoT technologies for FL. This discovery could hasten the FL and vehicular IoT development. The multi-path notion is greatly impacted by the mathematical model in smart environments. It has been found that our suggested model outperforms all other models in smart environment-based multi-path scenarios.



(a) Transportation Cost Minimization

(b) Fuel cost minimization

(c) Parking cost minimization

(d) Safety value minimization

(e) Traffic cost minimization

Fig. 6. Cost minimization models comparison.

7.1. Limitations

FL-based vehicular networks can improve various aspects of smart cities, such as traffic management and road safety. They have some limitations to consider when implementing this technology:

The quality of connectivity between the participating devices in the vehicular network. In areas with limited connectivity, the accuracy and efficiency of the models may be compromised. FL relies on data sharing between devices in the vehicular network. There are privacy concerns related to collecting, storing, and sharing personal data, including location and travel patterns, which may make users hesitant to participate. In a vehicular network, the data are heterogeneous due to different vehicle types, traffic patterns, and road conditions, which may affect the performance of the models While FL improves vehicular network performance in smart cities, these limitations must be addressed to ensure that the technology is deployed effectively and

Table 2
Results of fuel cost minimization.

Algo.	[Node, Route, Traffic, Safety, Cost, Fuel cost, Parking Cost]	Total parking	Total distance	Total cost($)	Total fuel	Total traffic	Total safety
iGA Multi Path	[0, 0, 0.01, 0.18, 80.08, 87.45, 93.97]– [8, 0, 0.4, 0.34, 79.82, 78.15, 100.57]– [9, 2, 0.17, 0.08, 65.79, 78.42, 76.67]– [6, 1, 0.87, 0.25, 92.51, 74.19, 90.32]– [2, 2, 0.18, 0.08, 90.38, 83.38, 92.29]– [3, 0, 0.2, 0.24, 91.38, 81.61, 93.46] [7, 2, 0.09, 0.03, 80.49, 88.03, 90.02]– [5, 2, 0.54, 0.44, 72.74, 100.41, 83.63]– [4, 0, 0.86, 0.2, 81.34, 109.14, 85.11] [1, 2, 0.35, 0.27, 78.05, 97.33, 93.7]	878.11	102.29	812.58	899.74	3.67	2.11
ACO Multi Path	[0, 2, 0.96, 0.14, 65.84, 78.46, 73.14]– [1, 1, 0.69, 0.7, 70.86, 71.99, 93.84]– [4, 0, 0.3, 0.03, 75.89, 78.1, 81.15]- [8, 1, 0.12, 0.03, 73.9, 74.0, 79.92]– [6, 0, 0.58, 0.49, 70.14, 71.3, 93.03]– [9, 2, 0.28, 0.59, 62.67, 79.88, 74.57]– [5, 1, 0.97, 0.02, 78.55, 98.09, 119.79]– [3, 2, 0.23, 0.29, 76.78, 107.75, 85.81]– [7, 1, 0.6, 0.14, 86.44, 106.87, 112.26]- [2, 0, 0.05, 0.6, 76.48, 100.32, 90.83]	866.76	92.05	737.55	904.34	4.78	3.03
iGA Single Path	[0, 1, 0.05, 0.59, 70.35, 80.6, 111.91]– [9, 2, 0.17, 0.08, 65.79, 79.3, 79.22]– [6, 2, 0.62, 0.13, 64.37, 77.72, 101.32]- [7, 1, 0.6, 0.14, 86.44, 75.39, 126.14]– [2, 2, 0.15, 0.07, 67.96, 0.0, 90.09]– [1, 2, 0.35, 0.67, 79.29, 77.15, 99.74]– [5, 0, 0.25, 0.26, 68.5, 103.75, 73.74]– [4, 1, 0.15, 0.52, 86.29, 96.2, 101.05]– [3, 1, 0.69, 0.28, 58.96, 96.0, 73.52]– [8, 0, 0.66, 0.06, 77.02, 104.27, 84.84]	790.38	92.77	724.97	941.57	3.69	2.8
ACO Single Path	[0, 2, 0.2, 0.54, 80.38, 74.83, 111.3]– [6, 0, 0.62, 0.98, 65.99, 84.12, 74.85]– [7, 0, 0.71, 0.68, 72.31, 88.59, 67.81]– [9, 2, 0.94, 0.04, 84.36, 70.68, 121.49]– [1, 1, 0.62, 0.14, 70.23, 0.0, 80.75] [8, 1, 0.47, 0.09, 81.28, 72.76, 125.53]- [3, 2, 0.05, 0.09, 92.23, 70.01, 95.88]– [2, 1, 0.65, 0.05, 87.09, 88.59, 104.45]– [4, 0, 0.22, 0.28, 98.9, 100.51, 136.09]– [5, 2, 0.05, 0.42, 77.07, 94.87, 74.6]	744.96	102.58	809.84	992.75	4.53	3.31

Table 3
Results of cost minimization.

Algo.	[Node, Route, Traffic, Safety, cost, fuel cost, Parking cost][Parking Cost]	Total parking	Total distance	Total cost($)	Total fuel	Total traffic	Total safety
iGA Multi Path	[0, 2, 0.66, 0.48, 89.95, 0.0, 95.52]– [9, 1, 0.29, 0.65, 64.24, 80.91, 87.74]– [4, 0, 0.3, 0.03, 75.89, 80.92, 103.75]- [8, 2, 0.08, 0.1, 89.51, 84.46, 101.95]– [7, 2, 0.64, 0.03, 88.49, 99.57, 92.63]– [6, 2, 0.76, 0.14, 77.0, 108.98, 73.63]– [3, 0, 0.77, 0.24, 77.39, 96.08, 98.23]– [2, 0, 0.15, 0.22, 66.84, 106.98, 75.2]– [5, 0, 0.2, 0.04, 92.28, 103.46, 100.9]– [1, 1, 0.68, 0.47, 62.16, 102.57, 91.6]	863.93	98.62	783.75	921.15	4.53	2.4
ACO Multi Path	[0, 2, 0.16, 0.3, 89.12, 87.21, 119.92]– [3, 1, 0.08, 0.37, 83.33, 76.36, 93.42]– [2, 2, 0.89, 0.13, 68.78, 83.78, 97.0]– [5, 0, 0.25, 0.26, 68.5, 87.19, 67.27]– [4, 2, 0.3, 0.27, 82.95, 75.88, 136.28]- [1, 2, 0.98, 0.13, 72.61, 78.29, 78.95]– [7, 1, 0.27, 0.01, 78.25, 106.32, 91.05]– [6, 1, 0.06, 0.77, 64.68, 100.53, 108.23]– [9, 0, 0.74, 0.09, 72.1, 101.59, 94.47]– [8, 0, 0.66, 0.06, 77.02, 92.68, 80.41]	889.83	98.59	757.34	967.0	4.39	2.39
iGA Single Path	[0, 0, 0.89, 0.17, 90.62, 83.57, 105.98]– [6, 2, 0.62, 0.13, 64.37, 74.62, 97.03]– [7, 2, 0.09, 0.03, 80.49, 89.89, 110.32]– [5, 2, 0.13, 0.93, 85.34, 82.07, 87.46]– [9, 2, 0.94, 0.04, 84.36, 70.68, 118.65]– [1, 0, 0.9, 0.15, 69.78, 84.45, 82.23]– [4, 0, 0.08, 0.01, 73.06, 79.31, 81.25]– [3, 2, 0.05, 0.09, 92.23, 106.62, 101.63]– [2, 1, 0.08, 0.74, 90.21, 100.42, 105.65]– [8, 0, 0.66, 0.06, 77.02, 94.0, 80.41]	865.63	103.09	807.48	970.61	4.44	2.35
ACO Single Path	[0, 0, 0.01, 0.18, 80.08, 89.14, 95.77]– [8, 0, 0.03, 0.42, 92.74, 70.52, 100.71]– [4, 1, 0.15, 0.52, 86.29, 70.09, 99.08]– [3, 2, 0.23, 0.29, 76.78, 80.79, 83.68]– [7, 2, 0.94, 0.21, 105.3, 71.64, 129.99]– [1, 0, 0.43, 0.79, 83.43, 73.78, 78.22]– [2, 0, 0.04, 0.36, 85.17, 100.87, 129.51]– [6, 1, 0.06, 0.77, 64.68, 93.47, 92.48]– [9, 2, 0.28, 0.59, 62.67, 101.91, 85.69]– [5, 2, 0.05, 0.42, 77.07, 104.6, 106.52]	856.81	99.38	814.21	1001.65	2.22	4.55

Table 4
Results of safety minimization.

Algo.	[Node, Route, Traffic, Safety, Cost, Fuel cost, Parking cost][Parking Cost]	Total parking	Total distance	Total cost($)	Total fuel	Total traffic	Total safety
iGA Multi Path	[0, 0, 0.01, 0.18, 80.08, 70.7, 100.78]– [8, 2, 0.7, 0.41, 93.14, 86.34, 100.12]– [2, 2, 0.16, 0.83, 87.17, 79.18, 89.58]– [7, 1, 0.12, 0.84, 94.03, 74.76, 89.47]– [3, 0, 0.95, 0.01, 66.78, 70.27, 71.23]– [9, 2, 0.94, 0.04, 84.36, 77.98, 129.27]– [1, 2, 0.35, 0.67, 79.29, 71.51, 94.1]– [5, 2, 0.13, 0.04, 94.49, 105.37, 93.85]– [6, 1, 0.1, 0.6, 82.65, 103.27, 103.32]– [4, 2, 0.05, 0.8, 65.81, 105.77, 74.04]	845.15	102.52	827.8	945.76	3.51	4.42
ACO Multi Path	[0, 0, 0.01, 0.18, 80.08, 76.16, 116.54]– [8, 2, 0.65, 0.74, 95.55, 72.82, 124.03]– [3, 1, 0.08, 0.37, 83.33, 108.65, 110.89]– [2, 2, 0.89, 0.13, 68.78, 78.08, 72.74]– [5, 1, 0.28, 0.82, 73.64, 101.87, 99.41]– [7, 1, 0.42, 0.35, 79.68, 93.21, 144.01]– [4, 2, 0.05, 0.68, 94.07, 97.79, 90.48]– [6, 2, 0.93, 0.89, 79.26, 96.8, 118.59]– [9, 2, 0.94, 0.04, 84.36, 93.07, 123.41]– [1, 2, 0.35, 0.27, 78.05, 92.99, 90.85]	911.44	103.86	816.8	1090.95	4.6	4.47
iGA Single Path	[0, 0, 0.01, 0.18, 80.08, 82.42, 102.71]– [8, 1, 0.56, 0.87, 64.56, 81.46, 83.92]– [4, 1, 0.22, 0.33, 68.09, 0.0, 82.84]– [5, 1, 0.23, 0.92, 86.84, 77.05, 96.29]– [2, 1, 0.9, 0.26, 76.64, 75.38, 84.46]– [7, 2, 0.64, 0.03, 88.49, 93.8, 116.24]– [6, 0, 0.61, 0.77, 85.4, 109.57, 88.78]– [3, 0, 0.95, 0.01, 66.78, 105.8, 94.3]– [9, 0, 0.14, 0.95, 62.56, 93.15, 74.81]– [1, 2, 0.35, 0.27, 78.05, 109.01, 91.0]	827.64	95.34	757.49	915.35	4.61	4.59
ACO Single Path	[0, 0, 0.01, 0.18, 80.08, 78.25, 83.51]– [8, 2, 0.9, 0.07, 73.62, 88.35, 85.33]– [6, 2, 0.37, 0.18, 72.3, 80.38, 99.32]– [5, 2, 0.13, 0.93, 85.34, 89.57, 111.0]– [9, 2, 0.48, 0.97, 79.61, 79.11, 104.2]– [7, 2, 0.19, 0.14, 89.68, 107.45, 93.59]– [2, 1, 0.94, 0.8, 79.54, 107.58, 89.57]– [1, 0, 0.28, 0.25, 87.94, 99.72, 121.55]– [3, 2, 0.04, 0.66, 79.9, 101.77, 141.23]– [4, 1, 0.74, 0.49, 74.46, 108.48, 75.19]	940.66	100.46	802.47	1004.49	4.08	4.67

Table 5
Results of traffic minimization.

Algo.	[Node, Route, Traffic, Safety, Cost, Fuel Cost, Parking cost]	Total parking	Total distance	Total cost($)	Total fuel	Total traffic	Total safety
iGA Multi Path	[0, 2, 0.8, 0.01, 83.43, 0.0, 87.14]– [2, 0, 0.31, 0.66, 86.47, 74.28, 111.48]– [9, 0, 0.06, 0.84, 72.25, 79.45, 72.04]– [5, 2, 0.5, 0.88, 97.63, 84.51, 113.04]– [8, 0, 0.51, 0.15, 67.55, 80.39, 79.05]– [3, 2, 0.51, 0.93, 92.94, 79.73, 121.47]– [1, 0, 0.58, 0.6, 78.91, 79.26, 82.9]– [7, 0, 0.5, 0.89, 97.77, 103.08, 91.0]– [6, 1, 0.1, 0.6, 82.65, 102.93, 91.8]– [4, 0, 0.43, 0.59, 79.97, 91.66, 94.81]	775.29	105.44	839.57	944.73	4.3	6.15
ACO Multi Path	[0, 2, 0.8, 0.01, 83.43, 0.0, 92.51]– [2, 1, 0.22, 0.25, 94.03, 86.53, 109.66]– [6, 2, 0.05, 0.26, 69.72, 75.64, 70.51]– [8, 1, 0.74, 0.98, 65.32, 86.25, 77.73]– [7, 0, 0.24, 0.54, 86.21, 78.1, 122.65]– [3, 2, 0.25, 0.06, 75.15, 81.87, 101.84]– [9, 2, 0.13, 0.91, 80.62, 89.93, 93.21]– [4, 0, 0.22, 0.28, 98.9, 88.7, 122.53]– [5, 1, 0.97, 0.82, 84.55, 76.2, 107.05]– [1, 0, 0.89, 0.55, 81.01, 92.7, 93.92]	755.92	98.62	818.94	991.61	4.51	4.66
iGA Single Path	[0, 2, 0.8, 0.01, 83.43, 0.0, 100.72]– [2, 0, 0.15, 0.22, 66.84, 77.45, 98.77]– [5, 1, 0.35, 0.94, 59.93, 75.38, 79.56]– [8, 2, 0.97, 0.65, 94.7, 79.08, 106.98]– [4, 1, 0.1, 1.0, 96.14, 77.43, 95.18]– [9, 1, 0.52, 0.53, 71.03, 77.7, 83.5]– [6, 1, 0.68, 0.55, 84.03, 88.7, 101.38]– [3, 2, 0.51, 0.93, 92.94, 91.78, 99.69]– [1, 1, 0.12, 1.0, 90.2, 93.11, 102.18]– [7, 1, 0.75, 0.97, 89.36, 95.9, 105.45]	756.53	104.49	828.6	973.41	4.95	6.8
ACO Single Path	[0, 2, 0.8, 0.01, 83.43, 0.0, 86.58]– [2, 1, 0.65, 0.05, 87.09, 72.02, 135.79]– [4, 1, 0.32, 0.04, 78.89, 81.49, 90.75]– [8, 1, 0.12, 0.03, 73.9, 71.46, 70.04]– [6, 0, 0.58, 0.49, 70.14, 87.87, 94.45]– [9, 2, 0.71, 0.1, 82.98, 82.16, 97.94]– [3, 2, 0.51, 0.93, 92.94, 104.43, 100.22]– [1, 1, 0.16, 0.9, 64.44, 104.44, 90.41]– [5, 2, 0.79, 0.6, 65.61, 106.78, 70.58]– [7, 1, 0.75, 0.97, 89.36, 104.64, 98.68]	815.29	101.12	788.78	935.44	5.39	4.12

Table 6
Results of parking minimization.

Algo.	[Node, Route, Traffic, Safety, cost, fuel cost, Parking Cost]	Total parking	Total distance	Total cost($)	Total fuel	Total traffic	Total safety
iGA Multi Path	[0, 1, 0.08, 0.15, 67.43, 85.47, 87.59]– [6, 1, 0.68, 0.55, 84.03, 80.1, 92.64]– [3, 0, 0.24, 0.54, 67.71, 72.72, 67.37]– [8, 2, 0.39, 0.4, 70.61, 89.08, 87.88]– [1, 0, 0.43, 0.79, 83.43, 76.84, 79.35]– [2, 2, 0.89, 0.13, 68.78, 78.23, 87.95]– [5, 0, 0.48, 0.78, 94.24, 81.05, 103.21]– [9, 1, 0.89, 0.55, 57.08, 75.22, 89.64]– [7, 1, 0.42, 0.35, 79.68, 70.42, 128.77]– [4, 0, 0.43, 0.59, 79.97, 76.97, 128.62]	786.1	97.07	752.96	953.02	4.93	4.83
ACO Multi Path	[0, 2, 0.47, 0.1, 63.73, 75.26, 99.64]– [8, 2, 0.43, 0.81, 59.85, 82.23, 69.98]– [5, 1, 0.13, 0.7, 66.35, 80.43, 82.87]– [6, 2, 0.13, 0.22, 88.0, 72.53, 92.25]– [1, 0, 0.9, 0.15, 69.78, 84.45, 85.85]– [4, 1, 0.15, 0.52, 86.29, 83.22, 90.4]– [3, 0, 0.77, 0.24, 77.39, 74.21, 93.57]– [2, 1, 0.57, 0.78, 76.85, 81.68, 114.92]– [9, 1, 0.89, 0.55, 57.08, 96.75, 68.98]– [7, 0, 0.11, 0.56, 71.79, 108.09, 87.25]	838.85	93.25	717.11	885.71	4.55	4.63
iGA Single Path	[0, 0, 0.78, 0.37, 67.22, 75.93, 94.29]– [2, 1, 0.9, 0.26, 76.64, 73.43, 81.6]– [7, 1, 0.22, 0.49, 71.9, 79.57, 79.64]- [5, 1, 0.97, 0.02, 78.55, 72.03, 77.39]– [3, 1, 0.82, 0.25, 92.89, 89.15, 106.94]– [9, 0, 0.14, 0.95, 62.56, 86.09, 99.46]– [1, 1, 0.69, 0.7, 70.86, 76.6, 78.11]– [4, 1, 0.32, 0.04, 78.89, 102.32, 105.34], [8, 2, 0.9, 0.07, 73.62, 103.27, 72.25]– [6, 0, 0.53, 0.9, 64.3, 97.98, 68.03]	856.37	92.4	737.43	863.05	6.27	4.05
ACO Single Path	[0, 0, 0.53, 0.68, 82.61, 84.73, 102.14]– [1, 1, 0.24, 0.99, 60.57, 81.97, 68.97]– [2, 0, 0.95, 0.87, 65.53, 72.84, 75.19]– [8, 1, 0.12, 0.03, 73.9, 82.78, 71.88]– [6, 2, 0.37, 0.18, 72.3, 78.47, 123.73]– [5, 1, 0.97, 0.02, 78.55, 77.57, 77.17]– [3, 2, 0.04, 0.66, 79.9, 107.93, 137.88]– [4, 2, 0.8, 0.86, 72.58, 108.51, 80.89]– [9, 1, 0.89, 0.55, 57.08, 109.79, 72.98]– [7, 2, 0.04, 0.6, 70.35, 109.96, 80.65]	914.55	92.66	713.37	891.48	4.95	5.44

Table 7
Results of all minimization.

Env.	Algo.	[Node, Route, Traffic, Safety, Cost, Fuel cost, Parking Cost]	Total parking	Total distance	Total cost($)	Total fuel	Total traffic	Total safety
Normal	iGA Multi Path	[0, 1, 0.05, 0.59, 70.35, 74.72, 111.84]– [9, 1, 0.89, 0.55, 57.08, 85.59, 67.81]– [7, 2, 0.52, 0.25, 89.47, 80.25, 84.5]– [4, 1, 0.12, 0.86, 73.64, 85.3, 112.04]– [2, 0, 0.04, 0.36, 85.17, 107.65, 92.49]– [6, 0, 0.53, 0.59, 68.43, 91.42, 77.28]– [8, 1, 0.48, 0.5, 64.92, 91.71, 73.86]– [1, 1, 0.16, 0.9, 64.44, 107.55, 77.39]– [5, 1, 0.97, 0.02, 78.55, 104.99, 77.42]– [3, 1, 0.68, 0.76, 69.89, 109.22, 105.2]	938.4	92.74	721.94	879.83	4.44	5.38
Smart	ACO Multi Path	[0, 2, 0.37, 0.39, 87.78, 71.47, 129.62]– [4, 1, 0.32, 0.04, 78.89, 77.24, 119.28]– [8, 0, 0.25, 0.16, 102.03, 82.61, 103.05]– [6, 1, 0.06, 0.77, 64.68, 83.92, 73.94]– [9, 2, 0.71, 0.1, 82.98, 82.05, 131.74]– [3, 0, 0.95, 0.48, 69.84, 86.78, 68.31]– [5, 0, 0.2, 0.04, 92.28, 82.23, 87.46]– [1, 2, 0.98, 0.13, 72.61, 94.86, 88.79]– [7, 2, 0.19, 0.14, 89.68, 103.41, 120.94]– [2, 0, 0.05, 0.6, 76.48, 109.88, 113.15]	874.45	102.8	817.25	1036.28	4.08	2.85
Normal	iGA Multi Path	[0, 1, 0.08, 0.15, 67.43, 72.03, 83.32]– [6, 1, 0.87, 0.25, 92.51, 87.85, 113.63]– [2, 0, 0.15, 0.22, 66.84, 71.06, 104.86]– [5, 0, 0.2, 0.04, 92.28, 72.13, 109.09]– [1, 1, 0.69, 0.27, 105.73, 82.27, 117.05]– [9, 1, 0.89, 0.55, 57.08, 73.34, 76.05]– [7, 0, 0.05, 0.26, 94.15, 76.56, 99.17]– [4, 2, 0.57, 0.15, 94.37, 97.5, 138.98]– [8, 0, 0.51, 0.15, 67.55, 98.81, 111.38]– [3, 2, 0.75, 0.18, 83.9, 99.96, 89.73]	831.51	100.74	821.84	1043.26	4.76	2.22
Smart	ACO Multi Path	[0, 0, 0.01, 0.18, 80.08, 87.04, 80.6]– [8, 1, 0.1, 0.43, 70.4, 81.24, 68.43]– [5, 1, 0.28, 0.82, 73.64, 89.1, 87.37]– [7, 2, 0.38, 0.29, 67.05, 84.04, 78.3]– [9, 1, 0.29, 0.65, 64.24, 71.44, 83.0]– [4, 0, 0.86, 0.2, 81.34, 71.74, 116.19]– [1, 1, 0.53, 0.41, 78.45, 0.0, 95.07]– [3, 2, 0.05, 0.09, 92.23, 86.72, 129.25]– [2, 2, 0.12, 0.93, 67.83, 83.57, 111.87]– [6, 1, 0.65, 0.94, 86.04, 0.0, 105.25]	654.89	97.24	761.3	955.33	3.27	4.94
Normal	iGA Single Path	[0, 0, 0.01, 0.18, 80.08, 85.57, 98.84]– [8, 0, 0.4, 0.34, 79.82, 86.39, 99.02]– [9, 0, 0.14, 0.95, 62.56, 77.73, 90.49]– [1, 0, 0.28, 0.25, 87.94, 85.24, 118.95]– [3, 0, 0.13, 0.97, 101.77, 94.18, 131.74]– [4, 0, 0.22, 0.28, 98.9, 102.38, 106.24]– [5, 0, 0.1, 0.76, 83.84, 91.04, 107.53]– [6, 0, 0.25, 0.46, 66.91, 104.11, 74.34]– [2, 0, 0.35, 0.54, 65.36, 98.68, 98.6]– [7, 0, 0.11, 0.56, 71.79, 107.97, 81.33]	933.29	99.13	798.97	1007.08	1.99	5.29
Smart	ACO Single Path	[0, 0, 0.01, 0.18, 80.08, 71.98, 82.66]– [8, 0, 0.09, 0.57, 93.77, 85.18, 97.18]– [2, 0, 0.15, 0.22, 66.84, 84.64, 107.94]– [5, 0, 0.1, 0.76, 83.84, 72.36, 118.14]– [6, 0, 0.19, 0.16, 78.8, 0.0, 140.45]– [1, 0, 0.28, 0.25, 87.94, 93.24, 113.89]– [3, 0, 0.13, 0.97, 101.77, 105.22, 107.08]– [4, 0, 0.31, 0.24, 74.34, 106.11, 81.15]– [7, 0, 0.71, 0.68, 72.31, 106.37, 76.91]– [9, 0, 0.18, 0.95, 74.65, 93.7, 79.68]	818.8	99.2	814.34	1005.08	2.15	4.98
Normal	iGA Single Path	[0, 0, 0.36, 0.13, 91.33, 81.23, 91.66]– [4, 0, 0.31, 0.24, 74.34, 76.86, 86.87]– [7, 0, 0.33, 0.04, 81.05, 87.24, 98.52]– [1, 0, 0.06, 0.96, 72.21, 89.49, 108.21]– [8, 0, 0.25, 0.16, 102.03, 77.46, 139.67]– [6, 0, 0.58, 0.49, 70.14, 92.62, 71.47]– [9, 0, 0.45, 0.5, 100.62, 104.66, 126.89]– [2, 0, 0.15, 0.22, 66.84, 98.28, 83.5]– [5, 0, 0.04, 0.64, 93.05, 106.13, 111.56]– [3, 0, 0.08, 0.86, 89.73, 105.23, 108.37]	919.2	101.58	841.34	1026.72	2.61	4.24
Smart	ACO Single Path	[0, 0, 0.01, 0.18, 80.08, 83.38, 77.46]– [8, 0, 0.25, 0.16, 102.03, 87.48, 103.98]– [6, 0, 0.62, 0.98, 65.99, 80.42, 68.9]– [7, 0, 0.05, 0.26, 94.15, 89.41, 114.72]– [4, 0, 0.08, 0.01, 73.06, 74.54, 82.55]– [3, 0, 0.77, 0.24, 77.39, 85.94, 101.38]– [2, 0, 0.15, 0.22, 66.84, 94.49, 80.2]– [5, 0, 0.2, 0.04, 92.28, 100.62, 87.75]– [1, 0, 0.37, 0.26, 105.19, 100.98, 107.32]– [9, 0, 0.18, 0.95, 74.65, 106.75, 107.38]	904.01	100.68	831.66	931.64	2.68	3.3

responsibly. Model: The proposed model uses the multi-path concept, which is enhanced by using several paths and vehicles. Algorithm: For small-scale problems, the proposed method operates relatively quickly; nevertheless, for large-scale problems, improvement is required.

7.2. Future scope

FL-based vehicular networks have a promising future in smart cities, and several potential applications can be explored in the coming years. Some of the future scopes of this technology are:

FL is used to develop autonomous driving systems that survey data from multiple vehicles to improve their performance. Autonomous vehicles make more accurate decisions and improve road safety by integrating data from sensors, cameras, and other sources. FL predicts real-time traffic patterns by investigating data from multiple vehicles. This information optimizes traffic flow, reduces congestion, and improves road safety. FL analyzes data from multiple vehicles to identify and prioritize infrastructure maintenance needs. Maintenance crews respond more quickly and prevent accidents by detecting potholes and road cracks. FL improves emergency response times by examining data from multiple vehicles to identify the fastest and safest routes for emergency vehicles. This helps save lives and reduce property damage during emergencies. FL optimize energy use in smart cities by analyzing data from multiple vehicles to identify opportunities for reducing energy consumption. This helps reduce carbon emissions and promote sustainability. The future scope of FL-based vehicular networks in smart cities is vast, and many potential applications can be explored in the coming years. With the continued development of this technology, we can expect to see significant improvements in traffic management, road safety, and overall quality of life in smart cities. The model can be expanded to a 4D-TSP-type problem with several paths and vehicles. It can also be expanded with fuzzy values. For iGA, mutation in the

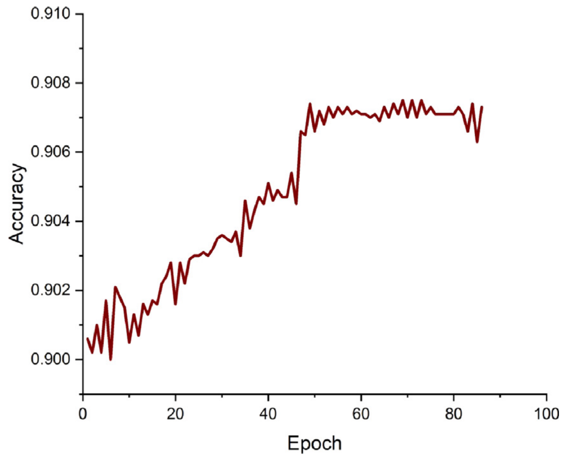


Fig. 7. Accuracy vs Epochs.

algorithm has been improved most, although crossover and selection operators can also be improved for greater performance.

CRediT authorship contribution statement

Sushovan Khatua: Conceptualization, Methodology, Software, Data curation, Visualization, Writing – original draft. **Debashis De:** Supervision, Validation, Writing – review & editing. **Somnath Maji:** Conceptualization, Methodology, Software, Visualization Writing – original draft. **Samir Maity:** Conceptualization, Methodology, Software, Data curation, Writing – review & editing. **Izabela Ewa Nielsen:** Supervision, Validation, Writing – review & editing.

Declaration of competing interest

We have no conflict of interest, mention all funding agencies and agree with the journal publishing ethics and rules.

Data availability

Data will be made available on request.

Acknowledgments

This research is supported by the AICTE Doctoral Fellow funded by the All India Council for Technical Education, India.

References

[1] Yaser Jararweh, Safa Otoum, Ismaeel Al Ridhawi, Trustworthy and sustainable smart city services at the edge, Sustainable Cities Soc. 62 (2020) 102394.

[2] Shiva Raj Pokhrel, Jinho Choi, A decentralized federated learning approach for connected autonomous vehicles, in: 2020 IEEE Wireless Communications and Networking Conference Workshops, WCNCW, IEEE, 2020, pp. 1–6.

[3] Zhaohua Zheng, Yize Zhou, Yilong Sun, Zhang Wang, Boyi Liu, Keqiu Li, Applications of federated learning in smart cities: Recent advances, taxonomy, and open challenges, Connect. Sci. 34 (1) (2022) 1–28.

[4] Sharnil Pandya, Gautam Srivastava, Rutvij Jhaveri, M. Rajasekhara Babu, Sweta Bhattacharya, Praveen Kumar Reddy Maddikunta, Spyridon Mastorakis, Md Jalil Piran, Thippa Reddy Gadekallu, Federated learning for smart cities: A comprehensive survey, Sustain. Energy Technol. Assess. 55 (2023) 102987.

[5] Yufeng Zhan, Peng Li, Zhihao Qu, Deze Zeng, Song Guo, A learning-based incentive mechanism for federated learning, IEEE Internet Things J. 7 (7) (2020) 6360–6368.

[6] Han Yu, Zelei Liu, Yang Liu, Tianjian Chen, Mingshu Cong, Xi Weng, Dusit Niyato, Qiang Yang, A fairness-aware incentive scheme for federated learning, in: Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, 2020, pp. 393–399.

[7] Shiva Raj Pokhrel, Jinho Choi, Improving TCP performance over WiFi for Internet of Vehicles: A federated learning approach, IEEE Trans. Veh. Technol. 69 (6) (2020) 6798–6802.

[8] Zisang Xu, Wei Liang, Kuan-Ching Li, Jianbo Xu, Hai Jin, A blockchain-based roadside unit-assisted authentication and key agreement protocol for Internet of Vehicles, J. Parallel Distrib. Comput. 149 (2021) 29–39.

[9] K. Hemant Kumar Reddy, Rajat Subhra Goswami, Diptendu Sinha Roy, IEEE Senior Member, A futuristic green service computing approach for smart city: A fog layered intelligent service management model for smart transport system, Comput. Commun. 212 (2023) 151–160.

[10] Ashish Rauniyar, Desta Haileselassie Hagos, Debesh Jha, Jan Erik Håkegård, Ulas Bagci, Danda B. Rawat, Vladimir Vlassov, Federated learning for medical applications: A taxonomy, current trends, challenges, and future research directions, 2022, arXiv preprint arXiv:2208.03392.

[11] Shiying Zhang, Jun Li, Long Shi, Ming Ding, Dinh C. Nguyen, Wuzheng Tan, Jian Weng, Zhu Han, Federated learning in intelligent transportation systems: Recent applications and open problems, IEEE Trans. Intell. Transp. Syst. (2023).

[12] Matteus Vargas Simão da Silva, Luiz Fernando Bittencourt, Adín Ramirez Rivera, Towards federated learning in edge computing for real-time traffic estimation in smart cities, in: Anais do IV workshop de Computação Urbana, SBC, 2020, pp. 166–177.

[13] Abdullatif Albaseer, Bekir Sait Ciftler, Mohamed Abdallah, Ala Al-Fuqaha, Exploiting unlabeled data in smart cities using federated learning, 2020, arXiv preprint arXiv:2001.04030.

[14] Nawaf Abdulla, Mehmet Demirci, Suat Ozdemir, Smart meter-based energy consumption forecasting for smart cities using adaptive federated learning, Sustain. Energy Grids Netw. 38 (2024) 101342.

[15] Mingzhe Chen, Zhaohui Yang, Walid Saad, Changchuan Yin, H. Vincent Poor, Shuguang Cui, A joint learning and communications framework for federated learning over wireless networks, IEEE Trans. Wireless Commun. 20 (1) (2020) 269–283.

[16] Mina Montazeri, Hamed Kebriaei, Babak N. Araabi, Jiawen Kang, Dusit Niyato, Distributed mechanism design in continuous space for federated learning over vehicular networks, IEEE Trans. Veh. Technol. (2022).

[17] Yinglong Li, Zhenjiang Zhang, Zhiyuan Zhang, Yi-Chih Kao, Secure federated learning with efficient communication in vehicle network, J. Internet Technol. 21 (7) (2020) 2075–2084.

[18] Huizi Xiao, Jun Zhao, Qingqi Pei, Jie Feng, Lei Liu, Weisong Shi, Vehicle selection and resource optimization for federated learning in vehicular edge computing, IEEE Trans. Intell. Transp. Syst. 23 (8) (2021) 11073–11087.

[19] Rodolfo Meneguette, Robson De Grande, Jo Ueyama, Geraldo P. Rocha Filho, Edmundo Madeira, Vehicular edge computing: Architecture, resource management, security, and challenges, ACM Comput. Surv. 55 (1) (2021) 1–46.

[20] Ufuk Dereci, Muhammed Erkan Karabekmez, The applications of multiple route optimization heuristics and meta-heuristic algorithms to solid waste transportation: A case study in Turkey, Decis. Anal. J. 4 (2022) 100113.

[21] Sabur Baidya, Yu-Jen Ku, Hengyu Zhao, Jishen Zhao, Sujit Dey, Vehicular and edge computing for emerging connected and autonomous vehicle applications, in: 2020 57th ACM/IEEE Design Automation Conference, DAC, IEEE, 2020, pp. 1–6.

[22] S.M. Ahsan Kazmi, Tri Nguyen Dang, Ibrar Yaqoob, Anselme Ndikumana, Ejaz Ahmed, Rasheed Hussain, Choong Seon Hong, Infotainment enabled smart cars: A joint communication, caching, and computation approach, IEEE Trans. Veh. Technol. 68 (9) (2019) 8408–8420.

[23] Junhua Wang, Kun Zhu, Ekram Hossain, Green Internet of Vehicles (IoV) in the 6G era: Toward sustainable vehicular communications and networking, IEEE Trans. Green Commun. Netw. 6 (1) (2021) 391–423.

[24] Sungwook Kim, Incentive design and differential privacy based federated learning: A mechanism design perspective, IEEE Access 8 (2020) 187317–187325.

[25] Viraaji Mothukuri, Reza M. Parizi, Seyedamin Pouriyeh, Yan Huang, Ali Dehghantanha, Gautam Srivastava, A survey on security and privacy of federated learning, Future Gener. Comput. Syst. 115 (2021) 619–640.

[26] Yu Wang, Guan Gui, Haris Gacanin, Bamidele Adebisi, Hikmet Sari, Fumiyuki Adachi, Federated learning for automatic modulation classification under class imbalance and varying noise condition, IEEE Trans. Cogn. Commun. Netw. 8 (1) (2021) 86–96.

[27] Monika Arya, Hanumat Sastry, Bhupesh Kumar Dewangan, Mohammad Khalid Imam Rahmani, Surbhi Bhatia, Abdul Wahab Muzaffar, Mariyam Aysha Bivi, Intruder detection in VANET data streams using federated learning for smart city environments, Electronics 12 (4) (2023) 894.

[28] S. Mahdi Homayouni, Dalila B.M.M. Fontes, José F. Gonçalves, A multistart biased random key genetic algorithm for the flexible job shop scheduling problem with transportation, Int. Trans. Oper. Res. 30 (2) (2023) 688–716.

[29] Somnath Maji, Samir Maity, Sumanta Bsau, Debasis Giri, Manoranjan Maiti, Varied offspring memetic algorithm with three parents for a realistic synchronized goods delivery and service problem, Soft Comput. 28 (5) (2024) 4235–4265.

[30] Yong Wang, Yuanhan Wei, Xiuwen Wang, Zheng Wang, Haizhong Wang, A clustering-based extended genetic algorithm for the multidepot vehicle routing problem with time windows and three-dimensional loading constraints, Appl. Soft Comput. 133 (2023) 109922.

[31] Anil Kumar Agrawal, Susheel Yadav, Amit Ambar Gupta, Suchit Pandey, A genetic algorithm model for optimizing vehicle routing problems with perishable products under time-window and quality requirements, Decis. Anal. J. 5 (2022) 100139.

[32] Sushovan Khatua, Anwesha Mukherjee, Debashis De, Sovec: Social vehicular edge computing-based optimum route selection, Veh. Commun. (2024) 100764.

[33] Huixia Cui, Jianlong Qiu, Jinde Cao, Ming Guo, Xiangyong Chen, Sergey Gorbachev, Route optimization in township logistics distribution considering customer satisfaction based on adaptive genetic algorithm, Math. Comput. Simulation 204 (2023) 28–42.

[34] Weihao Ouyang, Xiaohong Zhu, Meta-heuristic solver with parallel genetic algorithm framework in airline crew scheduling, Sustainability 15 (2) (2023) 1506.

[35] Ahmad Altarabsheh, Ibrahim Altarabsheh, Mario Ventresca, A hybrid genetic algorithm to maintain road networks using reliability theory, Struct. Infrastruct. Eng. 19 (6) (2023) 810–823.

[36] Tomoki Yasuda, Shinichi Ookawara, Shiro Yoshikawa, Hideyuki Matsumoto, Materials processing model-driven discovery framework for porous materials using machine learning and genetic algorithm: A focus on optimization of permeability and filtration efficiency, Chem. Eng. J. 453 (2023) 139540.

[37] Hemant Kumar Apat, Bibhudutta Sahoo, Veena Goswami, Rabindra K. Barik, A hybrid meta-heuristic algorithm for multi-objective IoT service placement in fog computing environments, Decis. Anal. J. 10 (2024) 100379.

[38] Xiaohui Wang, Xiaokun Zheng, Xiao Liang, Charging station recommendation for electric vehicle based on federated learning, J. Phys.: Conf. Ser. 1792 (1) (2021) 012055.

[39] Yuris Mulya Saputra, Diep N. Nguyen, Dinh Thai Hoang, Thang Xuan Vu, Eryk Dutkiewicz, Symeon Chatzinotas, Federated learning meets contract theory: energy-efficient framework for electric vehicle networks, 2020, arXiv preprint arXiv:2004.01828.

[40] Ferheen Ayaz, Zhengguo Sheng, Daxin Tian, Yong Liang Guan, A blockchain based federated learning for message dissemination in vehicular networks, IEEE Trans. Veh. Technol. 71 (2) (2021) 1927–1940.

[41] Atif Rizwan, Rashid Ahmad, Anam Nawaz Khan, Rongxu Xu, Do Hyeun Kim, Intelligent digital twin for federated learning in AIoT networks, Internet Things (2023) 100698.

[42] Ahmet M. Elbir, Burak Soner, Sinem Çöleri, Deniz Gündüz, Mehdi Bennis, Federated learning in vehicular networks, in: 2022 IEEE International Mediterranean Conference on Communications and Networking, MeditCom, IEEE, 2022, pp. 72–77.

[43] Prajwal Keshavamurthy, Emmanouil Pateromichelakis, Dirk Dahlhaus, Chan Zhou, Edge cloud-enabled radio resource management for co-operative automated driving, IEEE J. Sel. Areas Commun. 38 (7) (2020) 1515–1530.

[44] Zhaoyang Du, Celimuge Wu, Tsutomu Yoshinaga, Kok-Lim Alvin Yau, Yusheng Ji, Jie Li, Federated learning for vehicular Internet of Things: Recent advances and open issues, IEEE Open J. Comput. Soc. 1 (2020) 45–61.

[45] Narisu Cha, Zhaoyang Du, Celimuge Wu, Tsutomu Yoshinaga, Lei Zhong, Jing Ma, Fuqiang Liu, Yusheng Ji, Fuzzy logic based client selection for federated learning in vehicular networks, IEEE Open J. Comput. Soc. 3 (2022) 39–50.

[46] Kang Tan, Duncan Bremner, Julien Le Kernec, Muhammad Imran, Federated machine learning in vehicular networks: A summary of recent applications, in: 2020 International Conference on UK-China Emerging Technologies, UCET, IEEE, 2020, pp. 1–4.

[47] Senthil Kumar Swami Durai, Mary Divya Shamili, Smart farming using machine learning and deep learning techniques, Decis. Anal. J. 3 (2022) 100041.

[48] Arun Kumar Dey, Govind P. Gupta, Satya Prakash Sahu, A metaheuristic-based ensemble feature selection framework for cyber threat detection in IoT-enabled networks, Decis. Anal. J. 7 (2023) 100206.

[49] Ferran Adelantado, Majsa Ammouriova, Erika Herrera, Angel A. Juan, Swapnil Sadashiv Shinde, Daniele Tarchi, Internet of vehicles and real-time optimization algorithms: Concepts for vehicle networking in smart cities, Vehicles 4 (4) (2022) 1223–1245.

[50] Ayman A. Wazwaz, Khalid M. Amin, Noura A. Semari, Tamer F. Ghanem, Enhancing human activity recognition using features reduction in IoT edge and azure cloud, Decis. Anal. J. 8 (2023) 100282.

[51] K. Raghavendar, Isha Batra, Arun Malik, A robust resource allocation model for optimizing data skew and consumption rate in cloud-based IoT environments, Decis. Anal. J. 7 (2023) 100200.

[52] Yufan Sheng, A computational optimization research on ant colony optimization for the traveling salesman problem, J. Phys.: Conf. Ser. 2258 (1) (2022) 012006.