A Novel Homomorphic Blockchain Scheme for Intelligent Transport Services in Fog/Cloud and IoT Networks

Abdullah Lakhan, Tor-Morten Groenli, Huaming Wu, *Senior Member, IEEE*, Muhammad Younas, and George Ghinea, *Member, IEEE*

***Abstract*— Modern smart city services necessitate complex technological infrastructure with heterogeneous compute servers, networks, and communication protocols. However, there are many research issues in heterogeneous computing infrastructure for Intelligent transport systems (ITS) when using the services in the network. Therefore, the main objective of this paper is to intelligent transportation services and their underlying infrastructure, built on an amalgamation of the Internet of Things (IoT), cloud and fog computing, and associated technologies. Specifically, this paper investigates the challenging issues of security, processing costs, and communication delays that frequently occur during the communication of data and messages. We propose novel, secure, and cost-effective schemes based on blockchain-assisted homomorphic encryption techniques. A Secure, Cost-Optimal Workload Assignment (SCWA) algorithm and a blockchain scheme made possible by Partially Hashing Homomorphic Encryption and Decryption (PHHE/D) are designed to distribute workload efficiently. We developed a simulator, MOTEL, that simulates the different functions of the proposed schemes and all the necessary components. Using MOTEL and data sets from real transport companies, the proposed approach is tested and evaluated using various experiments. The results demonstrate that, compared to existing solutions, the proposed approach significantly reduces processing costs and delays while maintaining an appropriate level of security in transport services.**

***Index Terms*— Smart cities, intelligent transport services, blockchain, Internet of Things, fog/cloud computing.**

I. Introduction

SMART city is a broad concept that encapsulates various services for modernizing urban facilities such as transportation, healthcare, housing, local trade, and commerce. The proliferation of smartphones and other digital technologies plays a major role in current smart city practices, including the Internet of Things (IoT), cloud (fog/edge) computing, and the Internet. This paper explores and addresses the main challenges of security, processing cost, and delays, which are related to smart cities' intelligent transportation and the underlying digital infrastructure that is built on the combination of IoT, cloud and fog computing, and associated technologies such as sensors, accelerometers, GPS, and cameras.

In smart cities, IoT interconnects physical and digital devices and has been utilized in collecting data from various sources, which in turn enable data analytics and improve the operations and services of public transportation [1]. Similarly, cloud and fog computing provides an integrated infrastructure to seamlessly support different modes of transportation, such as subways, metros, trains, trams, and buses. Thus, based on the combination of IoT, cloud, and fog technologies, intelligent transportation systems can provide additional services such as timetables and schedules, traffic conditions, and real-time location updates based on data received from various sensors. IoT-enabled (beacon) sensors are vital in transmitting timely and contextual information about public transport modes to cloud/fog servers, which enable transport authorities and the general public to access desired information about transportation services [2]. In addition, mobile sensors, including accelerometers, gyroscopes, magnetometers, GPS, and cameras, are integrated with mobile devices to determine users' activities. These sensors collect valuable data that can be used for various purposes, such as automated ticketing, fare collection, and analyzing travel patterns across different modes of transportation [3].

Intelligent transports have various economic, efficiency, and environmental benefits. They assist people in planning journeys, prioritizing mobility, accessibility, and responsiveness to passengers' needs with reliable information [4]. They provide a platform wherein various public transport modes and their operators can collaborate to ensure seamless travel services [5], thus enabling passengers to utilize multiple transport modes [6]. This practice assists in planning journeys by providing accurate information and a wide range of services, including real-time scheduling updates, delays, route operations, and disruptions across different modes of transport. [2] Sharing data among public transportation providers

Received 28 December 2023; revised 24 July 2024; accepted 4 November 2024. This work was supported in part by the National Key Research and Development Program of China under Grant 31400, in part by the National Natural Science Foundation of China under Grant 62071327, and in part by Tianjin Science and Technology Planning Project under Grant 22ZYYYJC00020. The Associate Editor for this article was C. Long. *(Corresponding author: Huaming Wu.)*

Abdullah Lakhan and Tor-Morten Groenli are with the Ubiquitous Computing Technology Laboratory, Kristiania University College, 0107 Oslo, Norway (e-mail: Abdullahrazalakhan@gmail.com; Tor-Morten.Gronli@kristiania.no).

Huaming Wu is with the Center for Applied Mathematics, Tianjin University, Tianjin 300072, China (e-mail: whming@tju.edu.cn).

Muhammad Younas is with the School of Engineering, Computing and Mathematics, Oxford Brookes University, OX3 0BP Oxford, U.K. (e-mail: m.younas@brookes.ac.uk).

George Ghinea is with the Department of Computer Science, Brunel University of London, UB8 3PH London, U.K. (e-mail: george.ghinea@brunel.ac.uk).

This article has supplementary downloadable material available at https://doi.org/10.1109/TITS.2024.3493452, provided by the authors.

Digital Object Identifier 10.1109/TITS.2024.3493452

enables mobile applications and websites to access current information. Furthermore, it optimizes resource utilization and enables network patterns, demand, and resource capabilities identification through collaboration between operators. Data exchange between public transport modes is vital in developing ticketing and fare systems. Through this practice, public transport operators can facilitate smooth fare payments and ticket issuance across various modes of transportation [3]. The data analytics of different public transport modes helps to identify the vehicle, users, and data pattern to improve the future efficiency of the transport systems [7].

Efficient and reliable provisioning of the above services demands that the underlying digital infrastructure be secure and efficient and that processing time be minimized. For instance, current research studies [7], [8], [9] attempt to optimize delay-efficient constraints among connected base stations in transport services infrastructure. Similarly, offloading techniques have been deployed to minimize service delays among base stations. The studies in [10], [11], [12], and [13] propose cost-efficient scheduling schemes for transport applications in complex networks wherein many nodes are connected at the road-unit side of the infrastructure. Furthermore, the studies in [14], [15], and [16] considered security mechanisms for transport applications based on blockchain technologies and distributed fog cloud networks. Blockchain technology converts transport data into valid hashing based on different security algorithms (e.g., Advanced standard encryption (AES) and secure hashing (SHA-256)) in various frameworks. Though existing blockchain approaches [17], [18], [19], [20] follow different consensus algorithms, such as PoW (Proof of Work), proof of validation (PoV), proof of stake (PoS), and Byzantine fault-tolerant blockchain schemes, they suffer from long delays and high processing costs for transport applications in cloud and fog networks.

This paper presents novel, cost-effective, and secure homomorphic blockchain schemes for intelligent transport services using cloud, fog, and IoT networks. It exploits the Partial Homomorphic Blockchain Network Optimization Scheme for Consumer Electronics-Enabled Transport Workload Assignment. The goal is to optimize delays and processing costs and maintain an appropriate level of security for intelligent transport applications deployed across different networking and computing nodes, such as vehicle computing nodes, base stations, and fog nodes. The novelty and contributions of the proposed schemes are summarised as follows.

- **Partial Homomorphic Enabled Blockchain for Complex Networks**: The proposed scheme devises a blockchain-enabled secure complex network-aware framework in which data communication, offloading, and scheduling of requests are carried out securely and efficiently. It exploits the blockchain decentralized mechanism to connect autonomous computing nodes (e.g., vehicular devices, base stations, IoT, cloud, and fog nodes), which take part in the processing of functionality of intelligent transport applications. The paper makes a novel contribution to the work.
- **Heterogeneous Node Security Validation**: The scheme introduces homomorphic encryption-enabled hashing and PoW validation schemes for transferring data among different nodes. It also devises a secure, delay-efficient, cost-optimal workload assignment (SCWA) algorithm. Partially Hashing Homomorphic Encryption/Decryption (PHHE/D) encrypts data and requests at local devices and then offloads them to the fog for execution via base stations.
- **A novel and practical simulator called MOTEL-Transport**. It contains consumer electronics data, complex network node services, applications, and information relevant to base stations, fog, and cloud layers. It contends that existing literature needs to design a novel, practical simulator for intelligent transport applications.

The paper is organized as follows: Section II reviews related work. Section III presents the proposed approach and its architectural design. Section IV illustrates the Partially Hashing Homomorphic Encryption/Decryption (PHHE/D) algorithm, security, and workload assignment. Performance evaluation is presented in Section V. Section VI concludes the paper.

II. Related Work

This section discusses the existing intelligent transport systems for secure service management. We delve into their efforts in terms of methods, frameworks, and infrastructure, all closely related to our work. We define existing ITS systems' parameters (e.g., objectives, methods, infrastructure, and resources) and the proposed system in Table I.

Yang et al. [1] presented a vehicular ad-hoc network (VANET)-enabled vehicular services environment for random vehicle applications. The certificate aggregation encryption scheme (CASS) was introduced to provide secure services to different vehicles on the road unit side. The objective was to minimize the security risks of the distributed services in the VANET-based system. However, the prior work only focused on centralized and homogeneous node security. Huang and Zhi [4] proposed a blockchain-integrated MIMO-based scheme to improve this perspective, which allows secure communication between connected offloading and scheduling nodes based on blockchain schemes (e.g., proof of work and credibility). However, this work is limited as it only considers local nodes without mobility services for ITS applications. Prior studies [2], [3], [7] have suggested blockchain schemes such as proof of work and proof of validity for distributed nodes. These studies considered security constraints such as transactions and validation for vehicular applications. However, all nodes are uniform, and these studies do not consider heterogeneous nodes with different features. The AAKE-BIVT schemes suggested anonymous authentication keys, but they still suffered from unknown registration and sharing data privacy with unknown nodes. This means that any vehicle node can be added, and while data is secure, it is still part of blockchain nodes and shared with different nodes without considering privacy rules. To address this issue, feedback based on blockchain clap schemes is suggested in these studies [8], [9], [10], [11], [12]. The main goal is registering and making valid transactions based on offloading and scheduling schemes. Any vehicle registration is valid and only checks the data vehicle

TABLE I
EXISTING BLOCKCHAIN INTELLIGENT TRANSPORT SYSTEMS

Study	Security	Layer	Node	Methods	Vehicle Type	Simulator	Objective
[1]	Networking	Communication	Fog	CASS	Random	VANET	Security
[4]	Blockchain	Communication	Fog-Cloud	AES	Random	VANET	Security
[2], [3], [7], [8], [9], [10]	Blockchain	Comp.Comm	Edge-Cloud	SHA	Random	Cloud	Security
[13], [14], [15], [16], [17]	Blockchain	Computation	Edge-Cloud	AES	Random	MANET	Security
[18], [19], [20], [20], [21], [22]	Blockchain	Hy.Comp.	Edge-Cloud	AES	Random	Netsim++	Security
[23], [24], [25], [26], [27]	Blockchain	Hy.Comp.	Edge-Cloud	AES	Random	v2V	Security
Proposed	PHHE/D	Vehicle	Fog	Het.Vehicles	SCWA	MOTEL-Transport	Delay/Cost/Security

services in fog cloud networks. These studies distributed the services in fog and clouds and implemented them on the roadside unit.

However, the aforementioned studies suffered from resource issues when offloading all workloads of vehicle applications to the fog and cloud nodes. The prior studies only deployed and called the services without considering the resource issues in their blockchain schemes for vehicle services and applications. These studies [13], [14], [15], [16], [17], [18] suggested blockchain schemes based on joint offloading and resource scheduling constraints for vehicular services in fog and cloud networks. They proposed different resource allocation policies on mobile edge cloud and distributed fog cloud networks. These security, resource, and communication constraints are optimized for vehicular application services in decentralized fog cloud networks. However, the ratio of resource consumption is recorded as higher; still, all fog and cloud nodes suffer from load balancing issues in the ubiquitous services networks. The Internet of Things (IoT) sensors are connected to fog cloud services based on blockchain schemes, allowing local vehicles to communicate with the distributed servers for different services. These services include traffic prediction, transport timetable, ticket validation, road safety, speed calculation, and transport routing status in distributed networks. However, all studies exploited the advanced standard encryption (AES) and secure hashing algorithm (SHA-256) schemes [19], [20], [21], [22] for blockchain-based service validation among fog and cloud networks. These security mechanisms utilize symmetric and asymmetric mechanisms, where the hashing of information and services is offloaded and processed in different nodes. However, these AES and SHA schemes integrated into the blockchain performed different data validations in an immutable form based on proof of work and proof of credibility schemes for the transport services. Nevertheless, these services consume much more resources and processing time in transport blockchain schemes for transport services.

These studies [23], [24], [25], [26], [27] suggested homomorphic encryption-enabled blockchain schemes for different industrial Internet of Vehicle Things (IIoT) vehicular applications. The homomorphic schemes have various types, such as fully and partially homomorphic, with varying encryption schemes, such as ElGamal, Goldwasser-Micali, and Benaloh, to perform encryption on one node. The rest of the nodes perform computation on the encrypted data instead of plaintext, as done in previous AES and SHA-256 schemes. The main objective of these studies is to allow one node to encrypt and decrypt in the blockchain-based network and offload workloads from one node to another. These studies [28], [29], [30], [31], [32] suggested lightweight security protocols such as cryptography and hashing for intelligent transport system applications. These security analysis protocols validated the data transactions among different nodes. However, these studies could have avoided the higher cost of resources and processing time in blockchain-enabled transport services in mobile fog cloud networks.

III. The Proposed System and Architectural Design

This section describes the proposed system. First, it explains the architectural design and different scenarios of intelligent transport applications related to security and workload distribution. Second, it presents the system model and the associated mathematical notations and parameters used in the proposed scheme. Third, it illustrates bi-objective cost and response time optimization.

A. Architectural Design

The proposed system addresses the security and workload assignment challenges in intelligent transport (vehicular) applications deployed in complex distributed computing networks. These networks comprise IoT-enabled devices, base stations (BSs), fog and cloud computing networks, mobile users, and various transportation modes (buses, trams, trains, etc.). We approach this as a joint offloading and scheduling problem, considering heterogeneous computing nodes such as BSs, fog nodes, and cloud computing. The architecture, illustrated in Fig. 1, depicts the core components of an intelligent transport system: real-time data generation sensors, cooperative nodes with blockchain miners, public transport modes, and the overall network infrastructure encompassing BSs and other devices.

The proposed system is decentralized, employing a blockchain-based intelligent transport mechanism that utilizes partial homomorphic encryption and decryption. Each node ensures its security through blockchain technology. The ITS sensor devices offload their real-time data to the BSs, and then the BSs forward the requests to the available fog nodes for processing. Each vehicle sensor can access only one base station at a time, and many BSs are connected to each fog node. All the BSs and fog nodes can communicate with each other. The primary goal is to support the mobility features of the services for vehicular applications. The blockchain technology based on the ITS system is implemented at all fog nodes. Initially, the vehicular devices offload their workload to the centralized fog agent for processing via particular BSs. After processing data, the centralized fog node offloads the data to another fog node based on blockchain technology with attributes. For instance, a centralized fog node has a current hash, a previous hash, a partial homomorphic hashing encryption/decryption, a timestamp, and a PoW consensus. The data is offloaded from one fog node to another through hashing and processing without decryption, except at the centralized fog nodes. The base stations are transceivers, and WiFi connects to the base stations to communicate with the

Furthermore, the architecture depicts four main case scenarios:

1) Scenario 1: In this scenario, the vehicular applications are installed on the local devices, and offload and access services are distributed through fog cloud networks. The runtime environment of applications is the same as that of distributed fog cloud networks. For instance, we have implemented the X86 operating system-enabled cross-platform runtime [33] environment for the different IoT microservices and executed different vehicular applications on fog cloud networks. Mobile users exploit applications such as transport location services, ticketing, and ticket purchasing and validation based on real-time data generation. With the help of GPS and Bluetooth, these sensors can determine the location and routing zone and generate tickets for the end users for different transport modes (e.g., metro, bus, train, and tram). All the transport sensors offloaded data to the fog and cloud nodes based on secure blockchain validations. However, mobile users can invoke these services for location, routing, tickets, and trip planning in fog cloud networks. In case 1, all the vehicles and user devices are consensus nodes that offload their data based on the blockchain validation scheme and encrypt and decrypt based on a homomorphic scheme based on available resources. The traffic monitoring system is installed on local devices and generates requests for execution from the fog nodes via different BSs. The only workload requests and results are to encrypt and decrypt local devices based on partial homomorphic encryption. The encrypted hash data request for the particular workload is then offloaded to the BSs for further offloading to the fog nodes for processing. Each encrypted workload has a specific block inside blockchain-enabled fog cloud networks.

2) Scenario 2: The homomorphic-enabled hashed data from the end users and vehicle devices and the blockchain-enabled base stations must be validated in the second-case scenario. If the previous offloaded hashing data and current hashing data match at the base stations, it is annotated as valid data. The data was also validated among different base stations during the mobility of vehicles and end users in the distributed fog cloud vehicular environment. The scheduler checks the resource availability at the base stations and delays to validate the security among the base stations. The primary is that overloading and traffic prediction at the base stations must be controlled; otherwise, higher delays in offloading and service invoking could degrade the performance of applications in real-time data processing. For instance, real-time ticket validation can be done in different transport modes. Therefore, valid data and verification are received as receivers from different fog and cloud nodes at the base stations. All the base stations send and receive data with secure validation based on blockchain schemes for end users and vehicle applications.

3) Scenario 3: In this scenario, we show the mobility of services and the connection between distributed base stations. These base stations have a limited range; each is 1 kilometer from other base stations. The handoff technique connects one base station to another during the mobility of end users and the vehicle in the distributed environment of the architecture. The hand-off connectivity must ensure the security, delay, and deadline of the vehicle workload without degrading performance in the network. All the base stations are connected with the blockchain networks, where each hand-off of the base station also validates and transfers the secure requests of vehicles and services in a safe form.

4) Scenario 4: In this scenario, we offload the data between fog and cloud nodes for the storage and services involved. These fog and cloud nodes can share and run different applications' data based on the same runtime environment on X86. We validated the data transactions among nodes based on blockchain technology in an immutable form. The proposed system considers different devices and transport modes (such as trains, metros, buses, and cars) with potential applications of traffic detection, location searching, online ticket selling and purchasing, etc. The devices use local devices to encrypt and decrypt vehicular traffic, application data, and their results locally. To ensure appropriate security between vehicular devices and BSs (BSs), the proposed system devises a PoW scheme, which matches the hashing of requests before sending them to the fog nodes for processing. All the vehicles are connected to the BSs via different wireless technologies. The PoW scheme also ensures security between BSs and BSs and fog-to-fog during mobility. Locations 1 and 2 can transfer their data as encrypted data (encrypted transactions) based on the PoW scheme in the system. The users (passengers, etc.) can use Android vehicle applications to monitor the location and traffic inside vehicles during their trips.

In the proposed architecture, all vehicle applications are connected to different fog and cloud networks through a base station (mobile network) and WiFi, the transport Internet network. Vehicles should have access to wireless network services and IoT-related devices. It considers different processing nodes (e.g., vehicle devices, base stations, fog, and cloud nodes) and two computational nodes (fog and cloud nodes). The fog nodes, as shown in Fig. 1, are located at the road-unit side, and cloud computing is located multiple hops away from the devices in the network. The fog nodes offer services to the vehicle applications on the radio network.

Further, we design coarse-grained workloads of transport applications (e.g., trains, buses, and cars) and use such workloads to evaluate how the proposed architecture reduces workloads using a blockchain-enabled approach. The security

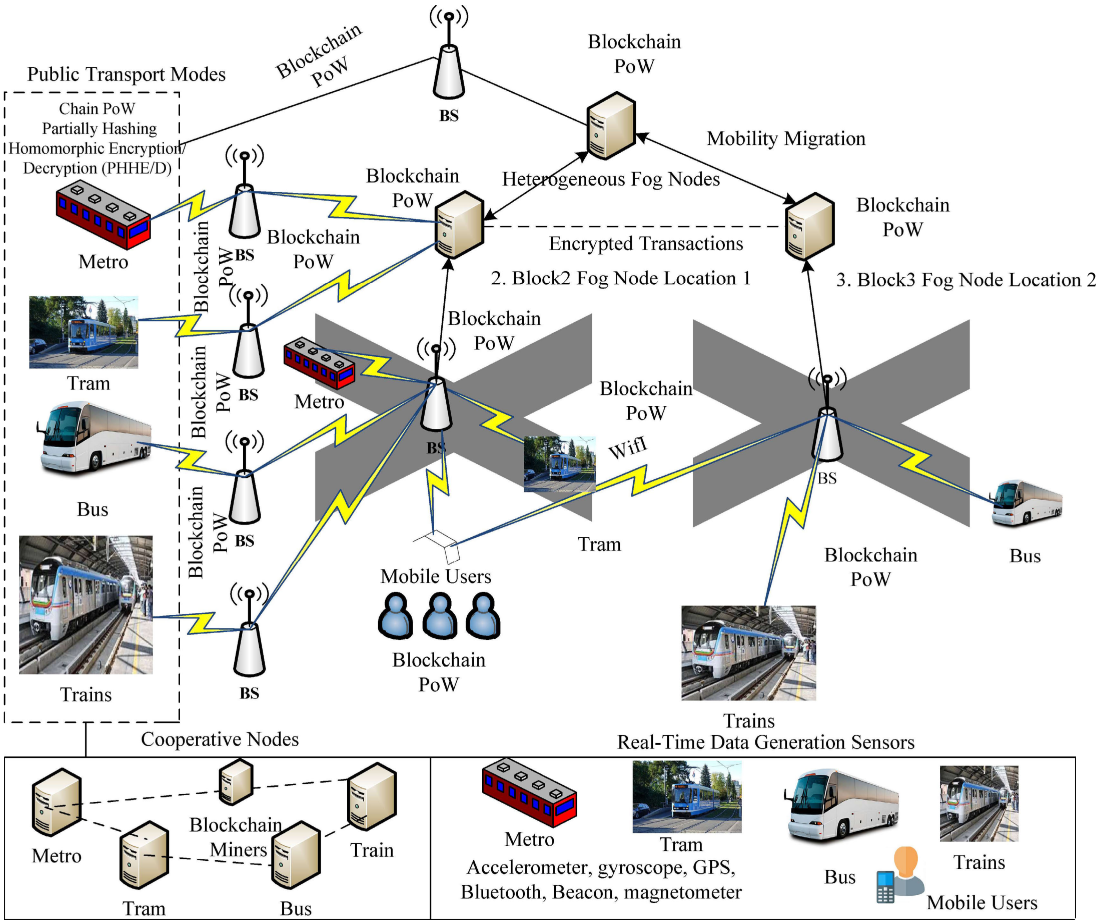


Fig. 1. Homomorphic secure and cost efficient blockchain system for public transport infrastructure and application.

mechanism ensures data is encrypted and decrypted on the secure nodes in the network.

B. Mathematical Modelling and Notations

This section presents a mathematical model of the proposed system. It also defines a set of mathematical notations (or parameters) used in developing and evaluating the proposed system.

Mobility is the key feature of vehicular applications, where a vehicle (e.g., bus or car) connects to all BSs by a certain distance in a distributed network. All the data and messages must be securely and efficiently communicated between respective devices and network nodes. The proposed system considers several important notations (or parameters) related to vehicular applications. Table II gives the notations of variables and their definition. The proposed model considers the V numbers of applications such as collision detection, traffic monitoring, etc. Each application v has a particular workload v_w and deadline v_d. It assumes that D number of vehicles and each vehicle d has resource ϵ_d and speed ζ_d. It also assumes the BS number of BSs and BSs has bs has limited resources ϵ_{bs} and speed ζ_{BS}. The model considers K number of homogeneous fog nodes with the same speed and resource, e.g., ζ_k and ϵ_k, respectively. The proposed model's workload assignments to different nodes are modeled as follows.

$$x_{v,bs,k} = \begin{cases} \dfrac{v_w}{Bw_{up\leftrightarrow Down}}, & x_{v,bs,k} = 1, \quad \textit{case 1} \\ \dfrac{v_w}{Bw_{up\leftrightarrow Down}}, & x_{v,bs,k} = 2, \quad \textit{case 2} \\ \dfrac{v_w}{Bw_{up\leftrightarrow Down}}, & x_{v,bs,k} = 3, \quad \textit{case 3} \\ \dfrac{v_w}{Bw_{up\leftrightarrow Down}}, & x_{v,bs,k} = 4, \quad \textit{case 4} \end{cases} \tag{1}$$

which determines the initial assignment of the workload in different cases. The proposed system makes the local encryption and decryption based on partial homomorphic and determines the time in the following way.

$$time_{LE} = \sum_{v_w=1}^{W}\sum_{v=1}^{V}\sum_{d=1}^{D} \frac{v_w}{\zeta_d} \times Enc + Dec \times x_{v,bs,k} = 1, \tag{2}$$

$$cost_{LE} = \sum_{v_w=1}^{W}\sum_{v=1}^{V}\sum_{d=1}^{w} \frac{v_w}{\zeta_d} \times Enc + Dec \times x_{v,bs,k} = 1, \tag{3}$$

where Eqs. (2) and (3) determine the local execution time and local cost of requests during encryption and decryption. However, the local time and cost are calculated after assignment on nodes as shown with this expression, e.g., $x_{v,bs,k} = 1$. The variable with assignment 1 shows the tasks assigned to the

TABLE II
MATHEMATICAL NOTATION

Problem Notations	Notation Definitions
V	Number of vehicular applications
v	The particular vehicular application v
W	Total number of workloads
v_w	Workload of application v
R	Total number of requests of V
r_w	Particular request of workload v_w
v_d	Deadline of vehicular application
D	Number of vehicle devices
d	Particular vehicle device
ζ_d	Processing speed of device d
ϵ_d	Resource of device d
BS	Number of BSs
bs	particular BSs
ζ_{bs}	Processing speed of BSs
ϵ_{bs}	Resource of BSs
K	Number of fog nodes
k	Particular fog node
ϵ_k	Resources of node k
B	Set of blockchain blocks
b	Particular block
p, q	Random number
PK, PV	Primary key, Private key 256-bits
$L1, L2, L3$	Initial Location, Mobility Location, Destination

nodes. Otherwise, it is equal to 0.

$$Enc \sim ciper = \sum_{v_w=1}^{W}\sum_{v=1}^{V}\sum_{d=1}^{D}\sum_{r=1}^{R}(PK, d, r \in v_w, p, q), \tag{4}$$

which determines the encryption process of all device requests with public and private keys. Both public and private keys are generated randomly based on random functions in the homomorphic operations in the system.

$$Dec = \sum_{v_w=1}^{W}\sum_{v=1}^{V}\sum_{d=1}^{D}\sum_{r=1}^{R} Enc(PV, d, r \in v_w, p, q), \tag{5}$$

which determines the encryption process of all requests on all devices with public and private keys. The execution time for particular fog nodes is determined as follows.

$$time_{FE} = \sum_{v_w=1}^{W}\sum_{v=1}^{V}\sum_{k=1}^{K}\frac{Enc \leftarrow v_w}{\zeta_k} \times x_{v,bs,k}, \tag{6}$$

$$cost_{FE} = \sum_{v_w=1}^{W}\sum_{v=1}^{V}\sum_{k=1}^{K}\frac{Enc \leftarrow v_w}{\zeta_k} \times x_{v,bs,k}. \tag{7}$$

Eqs. (6) and (7) determine the computation time and computation cost at fog nodes, respectively.

1) Case 1: Vehicular Devices to BSs (BSs): The offloading time between devices to BSs is determined in the following way.

$$time_{d\sim bs} = \sum_{bs=1}^{BS}\sum_{d=1}^{D}\frac{Enc \sim cipher}{Bw_{up\leftrightarrow Down}} \times \{x_{v,bs,k} = 1\}, \tag{8}$$

which determines the offloading time between devices and BSs. The offloading cost during request transmission from vehicle devices to BSs is determined in the following way.

$$cost_{d\sim bs} = \sum_{bs=1}^{BS}\sum_{d=1}^{D}\frac{Enc \sim cipher}{Bw_{up\leftrightarrow Down}} \times bandwidth \times cost \times \{x_{v,bs,k} = 1\}. \tag{9}$$

2) Case 2: BSs to Fog Nodes: The offloading time between devices to BSs is determined in the following way.

$$time_{bs\sim fog} = \sum_{bs=1}^{BS}\sum_{k=1}^{K}\frac{Enc \sim cipher}{Bw_{up\leftrightarrow Down}} \times \{x_{v,bs,k} = 2\}, \tag{10}$$

which determines the offloading time between BSs and fog nodes. The offloading time between BSs and fog nodes is determined in the following way.

$$cost_{bs\sim fog} = \sum_{bs=1}^{BS}\sum_{k=1}^{K}\frac{Enc \sim cipher}{Bw_{up\leftrightarrow Down}} \times bandwidth \times cost \times \{x_{v,bs,k} = 2\}, \tag{11}$$

which determines migration costs between BSs and fog nodes.

3) Case 3: Data Migration Between BSs: The offloading time between BSs and BSs is determined in the following way.

$$time_{bs\sim bs} = \sum_{bs=1}^{BS}\frac{Enc \sim cipher}{Bw_{up\leftrightarrow Down}} \times \{x_{v,bs,k} = 3\}, \tag{12}$$

which determines the offloading time between BSs and fog nodes. The offloading time between BSs and fog nodes is determined in the following way.

$$cost_{bs\sim bs} = \sum_{bs=1}^{BS}\frac{Enc \sim cipher}{Bw_{up\leftrightarrow Down}} \times bandwidth \times cost \times \{x_{v,bs,k} = 3\}, \tag{13}$$

which determines the migration cost between BSs and BSs.

4) Case 4: Fog to Fog: The offloading time between fog and fog is determined in the following way.

$$time_{fog\sim fog} = \sum_{bs=1}^{BS}\frac{Enc \sim cipher}{Bw_{up\leftrightarrow Down}} \times \{x_{v,bs,k} = 4\}, \tag{14}$$

which determines the offloading time between BSs and fog nodes. The offloading time between BSs and fog nodes is determined in the following way.

$$cost_{fog\sim fog} = \sum_{bs=1}^{BS}\frac{Enc \sim cipher}{Bw_{up\leftrightarrow Down}} \times bandwidth \times cost \times \{x_{v,bs,k} = 4\}, \tag{15}$$

which determines the migration cost between the fog node and the fog node.

C. Bi-Objective Cost and Response Time Optimization

The proposed architecture optimizes vehicular applications' response time and processing cost as a bi-objective optimization in distributed fog and cloud networks. The response time for applications is determined in the following way.

$$\begin{aligned} Total_{Response-Time} &= time_{LE} + time_{FE} + time_{d\sim bs} \\ &\quad + time_{bs\sim fog} + time_{fog\sim fog} + time_{bs\sim bs}, \end{aligned} \tag{16}$$

which determines the response time of all vehicular applications on fog nodes during processing and BSs during offloading.

$$\begin{aligned} Total_{Cost} &= cost_{LE} + cost_{FE} + cost_{d\sim bs} \\ &\quad + cost_{bs\sim fog} + cost_{fog\sim fog} + cost_{bs\sim bs}, \end{aligned} \tag{17}$$

which determines the cost of all vehicular applications on fog nodes during processing and BSs during offloading.

The objective is to minimize both the processing delay and the transportation costs associated with workload management for all applications, which can be defined as problem $\mathcal{P}$.

$$\mathcal{P} : \min \quad \{Total_{Response-Time}, Total_{Cost}\}, \tag{18}$$

$$s.t. \sum_{v_w=1}^{W}\sum_{v=1}^{V}\sum_{r=1}^{R} r \in rw \leftarrow v_w \leq v_d, \ \forall r = 1, \ldots R, \tag{19}$$

$$\sum_{v_w=1}^{W}\sum_{d=1}^{D}\sum_{v=1}^{V}\sum_{r=1}^{R} r \in rw \leftarrow v_w \leq \epsilon_d, \ \forall r = 1, \ldots, R, \tag{20}$$

$$\sum_{v_w=1}^{W}\sum_{bs=1}^{BS}\sum_{v=1}^{V}\sum_{r=1}^{R} r \in rw \leftarrow v_w \leq \epsilon_{bs}, \ \forall r = 1, \ldots, R, \tag{21}$$

$$\sum_{v_w=1}^{W}\sum_{k=1}^{K}\sum_{v=1}^{V}\sum_{r=1}^{R} r \in rw \leftarrow v_w \leq \epsilon_k, \ \forall r = 1, \ldots, R, \tag{22}$$

where Constraint (19) ensures that the deadlines of all vehicle applications must be met during both the request and processing phases, Constraint (20) determines that all devices have sufficient resources to run the workload, Constraint (21) determines that all BSs have sufficient resources to run the workload, and Constraint (21) determines that all fog nodes have sufficient resources to run the workload.

IV. Proposed Schemes and Algorithms

This section discusses the best ways to solve the problems of safe delay and cost-optimal workload assignment in architectural design (Fig. 1) and the system model cases we discussed in Section III. Accordingly, it devises a secure delay and cost-optimal workload assignment (SCWA) algorithm. It starts the Partially Hashing Homomorphic Encryption/Decryption (PHHE/D) process, which encrypts requests (data) from applications on local devices and sends them to the fog node for BSs to run. We present the SCWA algorithm, which ensures the processing of all workloads under their requirements. Algorithm 1 (of the framework) has different schemes, such as offloading schemes, scheduling schemes based on blockchain validation, workload assignment schemes, and homomorphic schemes that handle different workloads. In case 1, users and vehicle devices processed workloads locally based on homomorphic encryption and then offloaded them to the fog and cloud through base stations, as shown in case 2. The mobility of services for users and vehicles is considered in case 3. Case 4 shows the fog, cloud data migration, and collaboration in the distributed networks.

Algorithm 1 SCWA

```
Input : $D, V, v, v_w, BS, K$
begin
    foreach $(V\ as\ v)$ do
        Call PHHE/D Scheme to Encrypt and Decrypt Request Data;
        Call Offloading Scheme Between $D$ and $BSs$ as Case 1;
        Call Offloading scheme between $BSs$ and $K$ Case 2;
        Call Mobility and Handoff Scheme between $bs$ and $bs$ as Case3;
        Call Secure Data migration scheme between $k$ and $k$ as Case 4;
        Call Secure Delay and Cost Efficient Scheduling;
        Optimize $Total_{Cost}, Total_{Response-Time}$;
End process;
```

A. The PHHE/D Scheme Security Threat Model

We present the security threat model based on the PHHE/D scheme inside blockchain technology. The model is divided into two parts: partial holomorphic encryption and blockchain validation among nodes for vehicle data. We identified security attacks based on the designed pattern inside security; if the security pattern inside hashing is changed, we assume it is a security attack on the data. In this case, the node is scanned and verified until and unless it receives the original pattern of hashing inside the block.

For the security analysis, we match the hashing pattern in each blockchain block, which is available in the form of hashing. If the hashing pattern and offloaded hashed are matched, the system acknowledges it as authenticated and valid transactions. Otherwise, the transaction remains on hold unless the pattern is restored to its original form.

Our system considers heterogeneous computing nodes such as local transport, wireless, and edge cloud nodes. Therefore, when we apply blockchain technology on nodes, there are resource constraints issues in nodes. The main contribution of the PHHE/D method is to compute encryption and decryption on the resource-efficient nodes, and the resource constraint

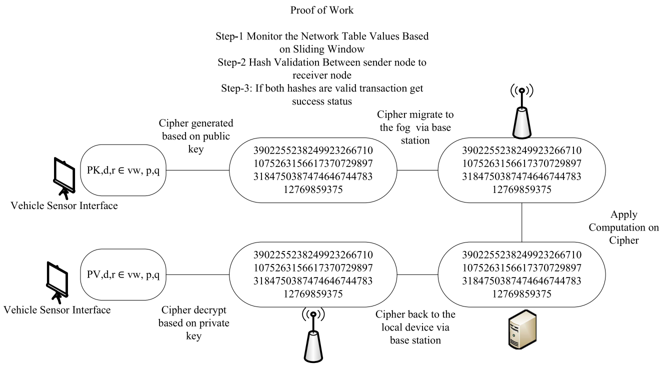

Proof of Work
Step-1 Monitor the Network Table Values Based on Sliding Window
Step-2 Hash Validation Between sender node to receiver node
Step-3: If both hashes are valid transaction get success status
Cipher generated based on public key
Cipher migrate to the fog via base station
PK,d,r ∈ vw, p,q
390225523824992326671010752631566173707298973184750387474646744783 12769859375
Vehicle Sensor Interface
Apply Computation on Cipher
PV,d,r ∈ vw, p,q
Vehicle Sensor Interface
Cipher decrypt based on private key
Cipher back to the local device via base station

Fig. 2. PoW scheme with homomorphic encryption.

computes the validation on the ciphertext instead of reprocessing the encryption and decryption. In this way, we can keep the balance between resource-constraint devices and rich resource nodes and maintain blockchain security among different nodes. Whereas existing PoW schemes perform the computation on all nodes with the same strategy that each node validates the data based on hashing and then starts the encryption and decryption for the next node, which is more time and resource consuming and resource constraints nodes cannot support this scenario for intelligent transport applications.

Therefore, we devise Partially Hashing Homomorphic Encryption/Decryption (PHHE/D) based on advanced standard encryption (AES). The scheme is diagrammatically represented in Fig. 2. The scheme exploits the asymmetric security mechanism, where the public key is used for encryption, and the private key is used for the resultant decryption. In the proposed scheme, local devices can encrypt data and results. Other devices can only transfer and apply computation to the data.

The goal of Algorithm 2 is to get partial homomorphic encryption on the transport workload using encryption and decryption models. The local transport devices can only encrypt and decrypt workload data based on partial homomorphic encryption and decryption and with pre-set homomorphic keys. The workload is encrypted based on the local device's AES 256-bit scheme. This is done in three phases. The workload is encrypted and decrypted with the additive mod function in the first phase. In the second phase, the iterative encryption and decryption process will continue until the iteration reaches the value n. In the third phase, the encryption and decryption are based on the mod function and n repetitions with the generated custom homomorphic key of 256 bits. This is represented in Fig. 3.

B. The PoW Scheme

The adaptive PoW scheme ensures blocks' data validity and authenticity based on different constraint rules. Algorithm 3 ensures and validates data originality based on generated encryption using previous and current hash. If the current node hash $k1$ is matched with the earlier node $k2$, the transaction gets a success status in the algorithm. However, in the case of a failure, the system will recall the procedure from the point of failure. We designed the offloading algorithm for different cases based on various constraints. For instance, availability of wireless bandwidth, signal capability, distance between end-user and vehicle devices and base stations, and hand-off parameters for vehicular applications. We implemented the distance formula between points [34] and Shannon theory [35] to offload the workloads under their deadlines. It impacts the overall cost of offloading in distributed fog cloud networks.

Algorithm 2 Partial Homomorphic Encryption and Decryption Scheme PHHE/D

```
Input : $v = 1, \ldots, V, v_w, \ldots V,\ D,\ K$
Output: Apply PHHE/D
begin
    Homomorphic-Key HK=[0101aabb];
    mod n;
    PHHE/D[$v, v_w, d, k$];
    foreach ($v = 1$ to $V$) do
        Compute Additive Encryption based on AES at local consumer devices ;
        Determine Encryption in Different Phases;
        Phase-1: Local devices computation;
        PHHE[$v, v_w, k$]= $\frac{\epsilon(v_w)}{\zeta_d}.\frac{\epsilon(v_w)}{\zeta_d}=\frac{\epsilon(v_w)}{\zeta_d} \times \frac{\epsilon(v_w)}{\zeta_k}$ mod $n$;
        Phase-2: Base Station Computation;
        Hashing pattern matched between local and base stations ;
        $(\frac{\epsilon(v_w)}{\zeta_{bs}} \times \frac{\epsilon(v_w)}{\zeta_{bs}})^e$ mod $n$;
        Phase-3:;
        $\epsilon(\frac{\epsilon(v_w)}{\zeta_k} \times \epsilon\frac{\epsilon(v_w)}{\zeta_k})^e$ mod $n$;
        Determined Decryption at local devices;
        PHHE/D[$v, v_w, k$]= $\frac{\epsilon(v_w)}{\zeta_k}.\frac{\epsilon(v_w)}{\zeta_k}=\frac{\epsilon(v_w)}{\zeta_k} \times \frac{\epsilon(v_w)}{\zeta_k}$ mod $n$;
        Phase-2:;
        Hashing pattern matched between base stations and edge nodes;
        $(\frac{\epsilon(v_w)}{\zeta_k} \times \frac{\epsilon(v_w)}{\zeta_k})^e$ mod $n$;
        Phase-3:;
        $\epsilon(\frac{\epsilon(v_w)}{\zeta_k} \times \epsilon\frac{\epsilon(v_w)}{\zeta_k})^e$ mod $n$;
        Call PoW Scheme based on Algorithm 3;
    End Inner encryption and decryption;
End Main loop;
```

C. The SCWA Scheme

This section devises the Secure Delay and Cost Optimal Workload Assignment (SCWA) scheme as shown in Algorithm 4. The goal is to handle four cases in which consumer electronics-enabled devices offload their workloads with encryption and decryption. In case 1, local vehicles and user devices enforce the security based on a homomorphic data scheme. In case 2, the scheme validated and verified hashing between vehicles and their connected BSs during offloading for data processing. The vehicle encrypts and decrypts workload locally based on blockchain attributes before processing and offloading them to the connected and associated BSs

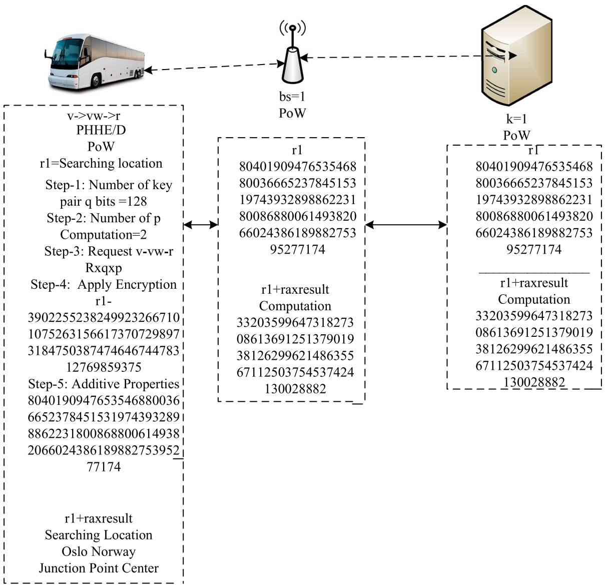


Fig. 3. PoW validation.

Algorithm 3 PoW Validation

```
Input : PHHE[$v, v_w, k$] $\in K, V$;
begin
    foreach ($k \leftarrow v_w$ to $K, V$) do
        Validated the offload hash between nodes;
        Declare CH Current Hash;
        Declare PH Previous Hash;
        foreach (f=1 as in M) do
            $f1 \leftarrow$ Tns1= Enc$\leftarrow$ [$v, v_w, k$] $\leftarrow SK$;
            $f2 \leftarrow$ Tns1= Dec $\leftarrow$ [$v, v_w, k$] $\leftarrow SK$;
            if (CH $\leftarrow$ Tns1 & PH$\leftarrow$ Tns1 ) then
                Immutable Shared transactions to another node;
                $k1 \leftarrow k2$;
            End Immutable Transactions;
        End Inner Transactions;
    End PoW;
```

for further processing and communication. In case 3, the scheme validated and verified the hashing between vehicles and their connected BSs during offloading for data processing. The vehicle encrypts and decrypts workload locally based on blockchain attributes before processing and offloading them to the connected and associated BSs for further processing and communication. In case 4, the scheme validated and verified the hashing between vehicles and their connected BSs during offloading for data processing. The vehicle encrypts and decrypts workload locally based on blockchain attributes before processing and offloading them to the connected and associated BSs for further processing and communication. Algorithm 4 offloads and allocates all workloads on the secure and optimal nodes for execution. The algorithm has two parts: secure offloading and optimal workload assignment under their resources and deadline, delay, and cost constraints.

Algorithm 4 SECURE DELAY AND COST OPTIMAL WORKLOAD ASSIGNMENT (SCWA) Scheme

```
Input : PHHE/D[$v, v_w, k$] $\in K, V, BS$;
begin
    foreach ($k \leftarrow v_w$ to $K, V$) do
        Determined initial assignment based on Eq. (1);
        For case-1;
        if ($\frac{v_w}{\zeta_k} \leq \epsilon_k$) then
            Call Algorithm 1;
            Determined the encryption time & decryption time based on Eqs. (4) and (5);
            Determined the local delay based on Eq. (2);
            $time_{LE} \leftarrow PHHE/D[v, v_w, k, bs1]$;
            Determined the cost based on Eq. (3);
            $time_{cost} \leftarrow PHHE/D[v, v_w, k, bs1]$;
            Call Algorithm 3;
        For case-2;
        if ($\frac{v_w}{\zeta_k} \leq \epsilon_{bs}$) then
            Determined the encryption time & decryption time based on Eqs. (4) and (5));
            Determined the local delay based on Eq. (10);
            $time_{bs \sim fog} \leftarrow PHHE/D[v, v_w, k, bs1] \leftarrow bs \sim PHHE/D[v, v_w, k, bs1] \leftarrow k$;
            Determined the cost based on Eq. (11);
            $cost_{bs \sim fog} \leftarrow PHHE/D[v, v_w, k, bs1] \leftarrow bs \sim PHHE/D[v, v_w, k, bs1] \leftarrow k$;
        For case-3;
        if ($\frac{v_w}{\zeta_k} \leq \epsilon_{bs}$) then
            Determined the encryption time & decryption time based on Eqs. (4) and (5);
            Determined the local delay based on Eq. (12);
            $time_{bs \sim bs} \leftarrow PHHE/D[v, v_w, k, bs1] \leftarrow bs \sim PHHE/D[v, v_w, k, bs1] \leftarrow bs$;
            Determined the cost based on Eq. (13);
            $cost_{bs \sim fog} \leftarrow PHHE/D[v, v_w, k, bs1] \leftarrow bs \sim PHHE/D[v, v_w, k, bs1] \leftarrow bs$;
        For case-4;
        if ($\frac{v_w}{\zeta_k} \leq \epsilon_{bs}$) then
            Determined the encryption time and decryption time based on Eqs. (4) and (5);
            Determined the local delay based on Eq. (14);
            $time_{bs \sim bs} \leftarrow PHHE/D[v, v_w, k, bs1] \leftarrow k \sim PHHE/D[v, v_w, k, bs1] \leftarrow bs$;
            Determined the cost based on Eq. (15);
            $cost_{bs \sim fog} \leftarrow PHHE/D[v, v_w, k, bs1] \leftarrow k \sim PHHE/D[v, v_w, k, bs1] \leftarrow bs$;
End Main;
```

Algorithm 4 is illustrated as follows.

1) The algorithm is devised to schedule all offloaded workloads on different computing nodes for processing. For instance, requests for ticket validation, searching timetable of vehicles, and location services can be

invoked from any available servers during the execution of different applications.

2) For case 1, the algorithm determines the security of offloaded data between local devices and base stations based on the blockchain scheme. The scheduler schedules the workloads locally before offloading them to fog and cloud nodes for further processing. In case 1, all the local processing and offloading are done based on minimum time and cost. The minimum cost and time are key constraints during local processing and offloading to the computing nodes for further analysis. For instance, if sufficient services and/or efficient wireless networks are unavailable, they consume more bandwidth and incur higher costs and time as they have to wait for the availability of services and data processing in the networks. The local offloaded engine in IoT devices ensures that wireless communications signals are weak and incur higher communication time. The offload load engine waits for the optimal communication channel with the optimal signals in the networks. Another aspect of an offload engine is that the fog and cloud nodes are busy and do not have free slots for new execution. Then, the local device engine method waits for offloading until fog and cloud slots are free for execution. Otherwise, end users and vehicle devices consume much more time and processing costs when local engines randomly offload workloads to fog and cloud nodes with weak signals and the unavailability of slots for processing. Therefore, the proposed scheme implements adaptive and optimal offloading based on the workload deadline and the urgency of the services in the networks.
3) For case 2, all the computing nodes, such as IoT, fog, and cloud computing nodes, are distributed and share their workloads to avoid resource scarcity and balance issues in our work. Another perspective is that we partitioned the applications into different computing nodes, such as fog and cloud, to meet the deadlines for workloads. The encrypted data and services are verified and validated based on the blockchain consensus scheme at the base stations. The PoW scheme validated the transactions of offloading and downloading data and services at different base stations. The proposed scheduler checks the available resources at base stations to validate the offloaded and downloaded data based on the given blockchain validation requirements. All base stations are assumed to have limited range and resources. We applied the handoff technique to offload the initial application requests to other base stations without degrading their performance. In case 2, the algorithm determined secure offloading between base stations and fog nodes based on blockchain validation, as we determined optimal processing time and cost (steps 14 to 19).
4) The algorithm determined the optimal hands-off and fog cloud scheduling in cases 3 and 4. Only original consensus nodes—such as local devices for data offloading and fog and cloud nodes for the services—handle the encryption and decryption. Therefore, we determined the algorithm's optimal time and processing cost (steps 20 to 34).
5) In all the steps mentioned above, Algorithm 4 ensures data processing security among different heterogeneous nodes. This shows that our proposed scheduler meets all the given constraints, such as security, delay, and deadline of applications on different computing nodes. It also satisfies all the cases mentioned above and their secure and valid data processing in different connected nodes.

Algorithm 4 optimizes the performances of applications in different cases. In different cases, we show the processing and offloading of application data on different nodes. The main goal of the scheduling scheme, as defined in Algorithm 4, is to process the data of applications into different cases with secure form and meet the given deadline requirements. Therefore, the process of each application is divided into different computing nodes, such as local devices, fog, and cloud nodes. Therefore, Algorithm 4 handles the scheduling mechanism of applications on different computing nodes under their given constraints.

V. Simulation and Evaluation

In this section, we compare existing systems with the proposed system. The main comparison is done based on their schemes for intelligent transport applications. We implemented the baseline approaches of existing system schemes, such as blockchain proof of work (POW) with the SHA-256 hashing schemes and advanced standard encryption (AES) with the proposed homomorphic schemes regarding processing time, cost, and deadlines.

A. Data Sources and Workload

We use datasets (as workload) collected from the Entur company in Oslo, Norway: https://developer.entur.org/stops-and-timetable-data. Entur offers many data-enabled services with different public transport modes. However, we consider only four public transport modes. Enter provides numerous datasets, such as transport timetables, stops, and traveler trips in various cities in Norway. We use Entur's Software Development Kits (SDK) to integrate these data services into our MOTEL simulator. However, we consider other aspects, such as traffic prediction and mobility services. Therefore, we collected further data from different Norwegian public transport data repositories. We gathered data for mobility services from the Kogenta company in Oslo, Norway. We integrated the SDK of Kogenta company to collect the data and used stored data from different users' devices and transport services for processing. We pre-trained users' data available at: https://www.kaggle.com/datasets/austcse/embedded-smartphone-sensor-data?resource=download. This user pattern data is implemented to track and determine the end users' patterns during public transport use in our Motel simulator.

B. MOTEL Transport Simulator

We have developed the MOTEL Transport Simulator, in which we implemented the SCWA and PHHE/D strategies

TABLE III
MOTEL Simulator Parameters

Value	Description
JAVA	Back up Development
Front End Developer	JAVASCRIPT
Data	XML Schema
Data and Applications	**Description**
v_{w1}	Metro Route and Services
v_{w2}	Tram Route and Services
v_{w3}	Bus Route and Services
v_{w4}	Trains Route and Services
$v1$	Transport Timetable
$v2$	Ticket Validation
$v3$	Traffic Information
$v4$	Mobility Services
D	32GB$\sim$ 64, 2GB$\sim$ 4GB Ram
K	5000GB$\sim$ 2TB, 32GB$\sim$ 128GB Ram
BS	100GB$\sim$ 200GB, 50Mbps$\sim$ 10Mbps
B	25
p, q, PK, PV	100,100, 256-bits

TABLE IV
End Users and Vehicle Data

ID	Time	Longitude	Latitude	Device	Speed	Sensor
1	6.00 am	-27.717	-1578	Bus	100	Many
2	7.00 am	-23.717	-2228	Metro	100	Many
3	8.00 am	-32.717	-3348	Bus	100	Many
4	9.00 am	-45.717	-4568	Train	100	Many
5	10.00 am	-345.717	-2358	Metro	100	Many
6	11.00 am	-22.717	-2348	Bus	100	Many
7	6.00 am	-23.717	-3458	Train	100	Many
8	12.00 pm	-112.717	-8978	Train	100	Many
9	13.00 pm	-1234.717	-7758	Train	100	Many
10	14.00 pm	-145.717	-4328	Bus	100	Many
11	15.00 am	-543.717	-4568	Bus	100	Many
12	16.00 am	-453.717	-1238	Bus	100	Many
13	17.00 am	-345.717	-3458	Tram	100	Many
14	18.00 am	-567.717	-4568	Tram	100	Many
15	22.00 am	-235.717	-23458	Tram	100	Many
5000	12.00 pm	-556.717	-4448	Tram	100	Many

to evaluate the performance of the proposed scheme with the given constraints such as cost, time, and deadlines. We defined the simulator parameters and elements in Table III.

The MOTEL Transport Simulator is developed to simulate the functions of all the necessary components, such as user applications, workloads and services, vehicle modes, IoT, fog, and cloud infrastructure.

The MOTEL simulator's applications are designed for smart devices with different services. In applications, we can search metro, tram, bus, and train schedules, traffic prediction, ticket validation, and trip planning based on different transport modes. There are two kinds of data in the MOTEL simulator: end-user data and vehicle and infrastructure data. We designed applications based on the X86 runtime environment, which supports any mobile applications and platforms in our simulation studies. The IoT sensor data such as accelerometer, magnetometer, gyroscope, GPS, and Bluetooth collects user and location data from vehicles and devices. We define this application in Table IV.

In the MOTEL simulator, with homomorphic-enabled blockchain technology, we have successfully integrated various cases to handle the complete processing of applications and resource management. These diverse cases encompass multiple scenarios and requirements, ensuring an efficient system for managing applications and resources. We have considered each case's specific needs in the implementation and tailored the blockchain solutions accordingly. We integrated the homomorphic encryption to convert plaintext into ciphertext and allow computations on ciphertext data without the prerequisite of prior decryption. The outcomes of these computations remain in encrypted form, and upon decryption, they yield an output that is indistinguishable from the result that would have materialized had the operations been executed on the unencrypted data. Utilizing homomorphic encryption holds significance in safeguarding privacy during endeavors related to outsourced storage and computation. This scheme facilitates the encryption of data and its transmission to cloud environments for processing, all while upholding its encrypted nature.

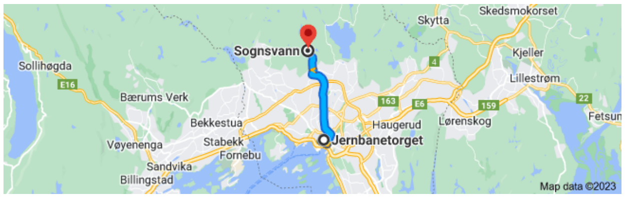


Fig. 4. Case 1 scenario during data collections.

1) Case Study 1: Homomorphic Secure Hashing Enabled Offloading Between Users-Vehicles and BSs: We implemented FHHED schemes based on homomorphic rules [36] with the additive operations. In MOTEL, the end users' devices, such as mobile machines, transport vehicle devices, and BSs, are considered computational nodes for generating and carrying data among different nodes. This means the basic encryption and decryption are performed locally on the user and vehicle devices. Each user and vehicle has many sensor types of data, e.g., accelerometer, magnetometer, gyroscope, GPS, and Bluetooth. These sensors collected real-time data generated by the integrated sensors in smartphones and vehicles, encompassing the accelerometer, gyroscope, magnetometer, battery sensor, GPS, etc. Such data was collected using a specialized Android application, Sensor Data Collector, explicitly designed to gather smartphone sensor data. The data help to determine the locations, mobility, patterns, and behavior of vehicles and end users for public data transport. The sensors also help to validate the issued transport tickets among different modes of public transport networks. For case 1, we implemented a practical environment of 15 kilometers in Oslo city from Janbantorget to Sognsvann, as shown in Fig. 4. We integrated Google Maps for location tracking with the GPS. There are BSs placed with a 1-kilometer difference among stations. The vehicles and end users' devices are connected to the BSs, where connections are established based on the handover technique [37].

Fig. 5 shows the offloading between users, vehicles, and associated BSs. Each BS is considered to have the resource capability to validate the offloading hashing with the original hash based on Algorithm 2 and Algorithm 3. Each sensor's data and service must be encrypted at the local user and

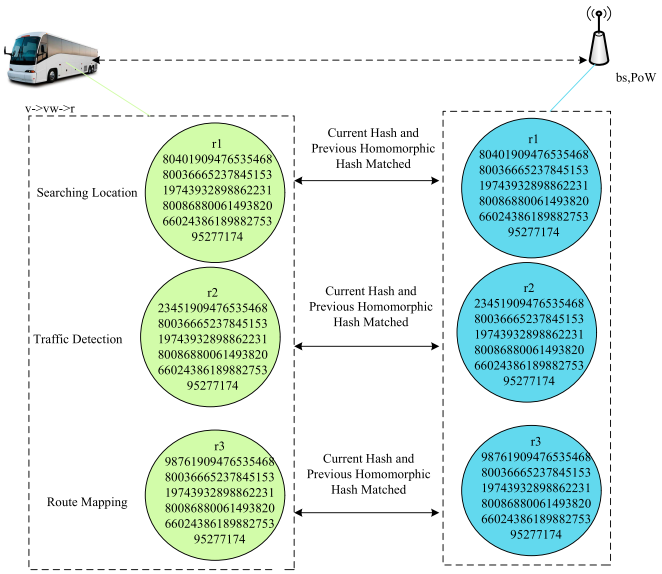


Fig. 5. Homomorphic encryption between vehicle and BSs.

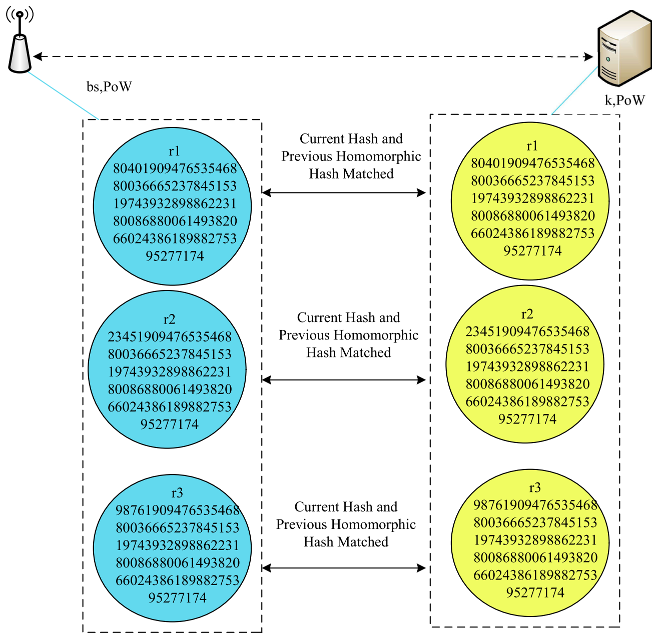


Fig. 6. Offloading between BSs and Fog Node.

vehicle devices before being offloaded to the BSs for further analysis. We validated the data offloading hashing based on the blockchain PoW scheme. For instance, the following expression designates a user inside the vehicle who is using a particular application: $v- > vw- > r$; whereas v is the location and traffic searching application, r is the location searching request, and vw is the sensors and vehicle workload, which are executed locally based on PHHED scheme and offload the encrypted data to the bs with the hashing, e.g., 80401909476535468. In it, the data receiving bs validated the offloaded data based on proof of validation and current hashing and then offloaded it to the available computing for further analysis.

2) Case Study 2: Homomorphic Secure Hashing Enabled Offloading Between BSs and Fog Nodes: In Case 2, the simulator shows offloading data from the BS onto the fog node. As discussed in Case 1, all the local users and vehicle devices offloaded their workloads to the base stations. Since the base stations serve as the communication medium between local end users and vehicle devices, we implemented blockchain schemes on each base station to validate and process the authenticated data sent to the fog nodes for processing. For instance, each base station bs implemented the blockchain PoW scheme, where offloaded requests (e.g., $r1 = 80401909476535468$) from local end users' devices must be validated using the previous hashing rules in blockchain blocks. The fog node k first validates the requested $r1 = 80401909476535468$ data (offloaded from bs) and then processes it. All the local devices, base stations, and fog nodes are part of the blockchain blocks for secure data validation. The blockchain validation process among nodes is the same for all offloaded request data, $r \in R$. Fig. 6 illustrates the offloading process between BSs and fog nodes for further processing. In our model, each base station has the resource capability to validate the offloaded hashing with the original hash based on Algorithm 2 and Algorithm 3.

3) Case Study 3: Homomorphic Secure Hashing Enabled Offloading Between BSs: In Case 3, we implemented a mobility mechanism for end users and vehicles in our simulator, which enables them to use different base stations for the same request. For instance, a mobility scenario is considered wherein a user moves from one location to another and covers a significant distance by walking while using location-searching services from any available server. Since each base station (bs) has a limited range, we designed a handover mechanism in the simulator. This mechanism allows base stations to automatically connect with servers and transfer their connections among them for data requests. To ensure secure request transfer, our blockchain scheme verifies the validity of base station switching for a particular request based on blockchain PoW schemes while maintaining performance and preventing tampering issues. Fig. 7 shows the offloading between BSs. In our model, each base station has the resource capability to validate the offloading hashing with the original hash based on Algorithm 2 and Algorithm 3. Therefore, the same request between base stations, e.g., $bs.r180401909476535468 \sim bs.r180401909476535468$ verified and validated blockchain PoW scheme. Thus, handoff among wireless channels is more secure in the mobility environment.

4) Case Study 4: Homomorphic Secure Hashing Enabled Offloading Between Fog Nodes: In this case, we demonstrate that the MOTEL simulator enables node mobility, data, and resource-balancing organization. For instance, a user can buy a ticket from one server and store their data on different servers for validation. This is because we consider different modes of transport with a single-ticket application. When a user purchases a transport ticket, it can be utilized for various modes, such as bus, taxi, metro, and train. Therefore, the MOTEL simulator facilitates migration among nodes, as depicted by $k \sim k$ in the scenario, where $r1 = 80401909476535468$.

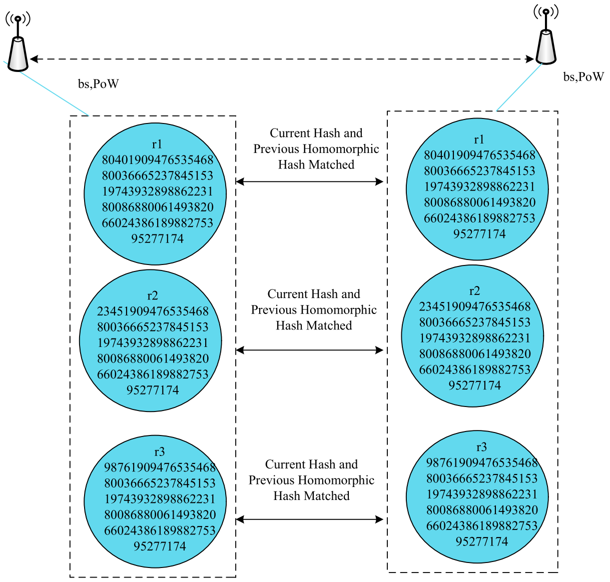


Fig. 7. Offloading between BSs.

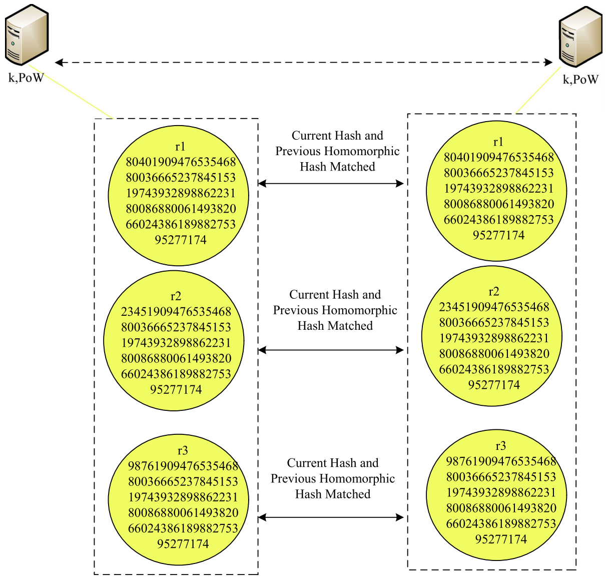


Fig. 8. Offloading between fog nodes.

All transactions for all requests are validated based on the blockchain PoW scheme. Fig. 8 illustrates the offloading between fog nodes. In our model, each base station has the resource capability to validate the offloading hashing with the original hash, following Algorithm 2 and Algorithm 3.

C. Results and Analysis

Fig. 9 shows the performance of blockchain schemes regarding the number of blocks, workloads, and mined transactions for all nodes. It can be seen from Fig. 9 that SCWA outperformed all blockchain schemes in terms of processing time.

We implement the Baseline Approaches for Heterogeneous Nodes [13], [14], [15], [19] and Baseline Approaches for Homogeneous Nodes [15], [17], [20], [21]. We analyzed and matched the results of each algorithm and technique based on four given cases, as stated above. In the first case, encryption and decryption are carried out on the local transport devices and offloaded to the BSs for further processing. The main reason is that local devices have resource constraints and cannot execute all workloads locally on the devices. Therefore, to improve performance, the vehicle offloads ciphertext to the BSs for further processing. In the second case, the cipher data was offloaded between the BSs and the fog node for processing. In the third case, the data was offloaded between fog nodes. In the fourth case, the data was offloaded between BSs.

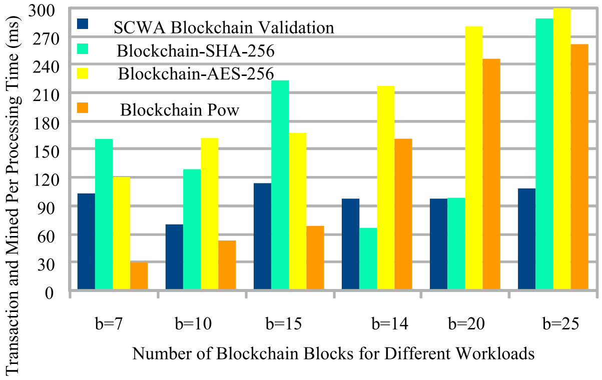
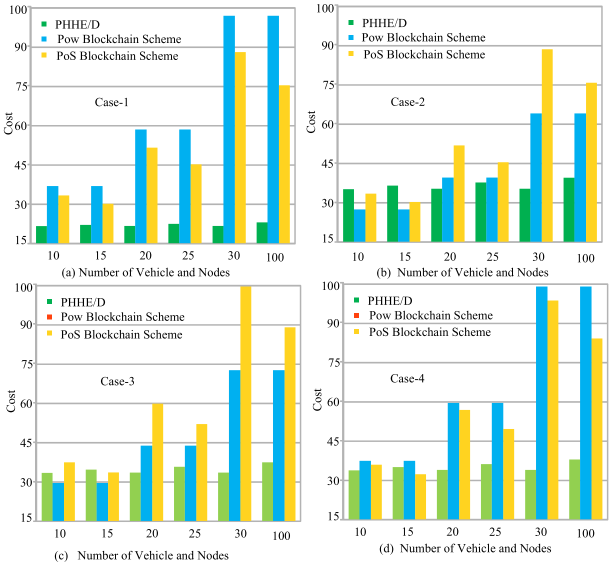


Fig. 9. Processing time of blockchain schemes for mined-transactions and blockchain validation with all workloads in nodes.



(a) Number of Vehicle and Nodes

(b) Number of Vehicle and Nodes

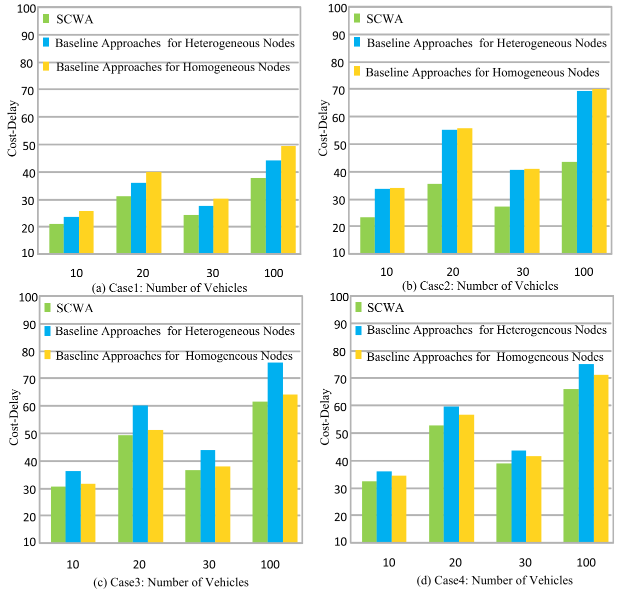
(c) Number of Vehicle and Nodes

(d) Number of Vehicle and Nodes

Fig. 10. Processing cost of algorithms and systems for vehicles applications.

Fig. 10 shows that the proposed PHHE/D scheme has a lower processing cost in four cases as compared to the existing PoW and proof of validation stake. This is the consequence of PoW and stake requiring encryption and decryption on each node, which is time-consuming. Similarly, as shown in Fig. 11, PoW and PoS require encryption and decryption on each node, which is time-consuming and requires many resources. Therefore, homomorphic encryption is time- and delay-optimal, as shown in Figs. 10 and 11.

Further, we conduct more experiments to compare our scheme with existing consumer electronics transport applications. The results are shown in Fig. 12. These show that the proposed homomorphic encryption and decryption have



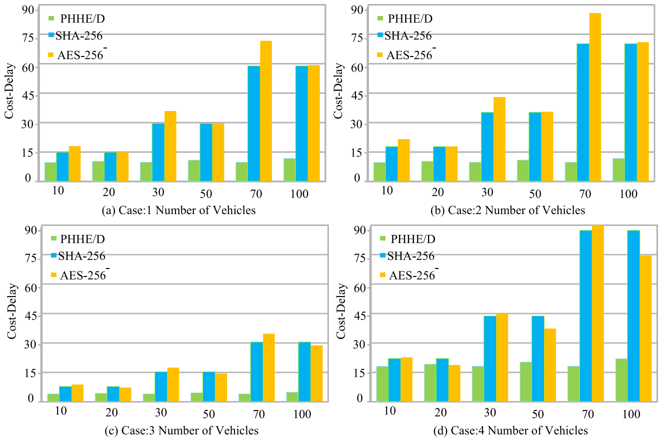
(a) Case1: Number of Vehicles

(b) Case2: Number of Vehicles

(c) Case3: Number of Vehicles

(d) Case4: Number of Vehicles

Fig. 11. Processing cost and delay of algorithms and systems for vehicles applications.



(a) Case:1 Number of Vehicles

(b) Case:2 Number of Vehicles

(c) Case:3 Number of Vehicles

(d) Case:4 Number of Vehicles

Fig. 12. Processing cost and delay of hashing and blockchain validation for vehicles applications.

a lower delay and cost than SHA-256 and AES, which are implemented in the existing blockchain for consumer electronics transport applications.

We implemented different encryption methods, such as generic cryptography and hashing techniques, which are widely used in blockchain technology to validate transport data with different services.

Fig. 13 shows the effectiveness of the proposed scheme PHHE/D compared to existing encryption techniques used in the blockchain for the validation of services of transport applications. PHHE/D has less processing time and cost in dollars than existing encryption methods. The main reason is that we perform the encryption computation on the rich resource node and do not repeat the encryption and decryption on each node for hashing, as the generally existing hashing and cryptography methods are done in blockchain technology. The partially homomorphic minimizes the time and cost to avoid re-encryption and decryption again and again on nodes and saves many resources consumption and service costs as shown in Fig. 13.

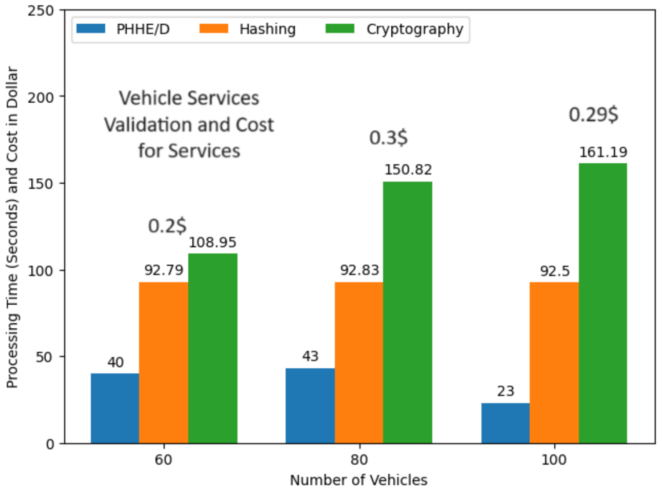


Fig. 13. Processing time and cost based on encryption methods for vehicle services.

VI. Conclusion

The work presented in this paper considers modern smart city technologies and applications in general. More specifically, it focused on the underlying infrastructure of intelligent transportation services built on state-of-the-art computing and networking technologies of IoT, cloud, fog, and mobile technologies. We designed, developed, and evaluated novel schemes of secure delay, cost-optimal workload assignment (SCWA), and partially hashing homomorphic encryption and decryption (PHHE/D). We designed various algorithms to implement the proposed schemes. The suggested plan starts with the PHHE/D-enabled blockchain plan, which encrypts the request data of consumer electronics-enabled transport apps on local devices and sends those apps' requests to the fog node to be run by BSs. We also developed a simulator called the MOTEL Transport Simulator. It simulates different functions of the proposed schemes and all the necessary components, such as user applications, workloads and services, vehicle modes, IoT, fog, and cloud infrastructure. We conducted various simulation experiments to evaluate the proposed schemes. Our schemes outperformed existing approaches, significantly reducing delays and processing costs and enhancing transport application security validation.

References

[1] Y. Yang, L. Zhang, Y. Zhao, K.-K.-R. Choo, and Y. Zhang, "Privacy-preserving aggregation-authentication scheme for safety warning system in fog-cloud based VANET," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 317–331, 2022.

[2] M. Haouari, M. Mhiri, M. El-Masri, and K. Al-Yafi, "A novel proof of useful work for a blockchain storing transportation transactions," *Inf. Process. Manage.*, vol. 59, no. 1, Jan. 2022, Art. no. 102749.

[3] E.-H. Diallo, O. Dib, N. R. Zema, and K. Al Agha, "When proof-of-work (PoW) based blockchain meets VANET environments," in *Proc. 12th Int. Conf. Inf. Commun. Syst. (ICICS)*, May 2021, pp. 336–343.

[4] Y. Huang and H. Zhi, "Blockchain-based transaction scheme for massive MIMO channel estimation," in *Proc. 11th Int. Conf. Netw., Commun. Comput.*, 2022, pp. 210–214.

[5] A. Tennøy, M. Knapskog, and F. Wolday, "Walking distances to public transport in smaller and larger Norwegian cities," *Transp. Res. D, Transp. Environ.*, vol. 103, Feb. 2022, Art. no. 103169.

[6] E. B. Lunke, "Modal accessibility disparities and transport poverty in the Oslo region," *Transp. Res. D, Transp. Environ.*, vol. 103, Feb. 2022, Art. no. 103171.

[7] A. Badshah et al., "AAKE-BIVT: Anonymous authenticated key exchange scheme for blockchain-enabled Internet of Vehicles in smart transportation," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 1739–1755, Feb. 2023.

[8] Y. Ren, F. Zhu, J. Wang, P. K. Sharma, and U. Ghosh, "Novel vote scheme for decision-making feedback based on blockchain in Internet of Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1639–1648, Feb. 2022.

[9] D. Chattaraj, B. Bera, A. K. Das, S. Saha, P. Lorenz, and Y. Park, "Block-CLAP: Blockchain-assisted certificateless key agreement protocol for Internet of Vehicles in smart transportation," *IEEE Trans. Veh. Technol.*, vol. 70, no. 8, pp. 8092–8107, Aug. 2021.

[10] S. Xia, Z. Yao, G. Wu, and Y. Li, "Distributed offloading for cooperative intelligent transportation under heterogeneous networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 16701–16714, Sep. 2022.

[11] X. Lin, Y. Li, J. Shao, and Y. Li, "Storage-assisted optical upstream transport scheme for task offloading in multi-access edge computing," *J. Opt. Commun. Netw.*, vol. 14, no. 3, pp. 140–152, Mar. 2022.

[12] S. Tong, Y. Liu, J. Mišić, X. Chang, Z. Zhang, and C. Wang, "Joint task offloading and resource allocation for fog-based intelligent transportation systems: A UAV-enabled multi-hop collaboration paradigm," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 11, pp. 12933–12948, Nov. 2023.

[13] A. Belogaev, A. Elokhin, A. Krasilov, E. Khorov, and Ian F. Akyildiz, "Cost-effective V2X task offloading in MEC-assisted intelligent transportation systems," *IEEE Access*, vol. 8, pp. 169010–169023, 2020.

[14] A. Balasubramaniam, M. J. J. Gul, V. G. Menon, and A. Paul, "Blockchain for intelligent transport system," *IETE Tech. Rev.*, vol. 38, no. 4, pp. 438–449, 2021.

[15] M. Humayun, N. Jhanjhi, B. Hamid, and G. Ahmed, "Emerging smart logistics and transportation using IoT and blockchain," *IEEE Internet Things Mag.*, vol. 3, no. 2, pp. 58–62, Jun. 2020.

[16] M. Oudani, "A combined multi-objective multi criteria approach for blockchain-based synchromodal transportation," *Comput. Ind. Eng.*, vol. 176, Feb. 2023, Art. no. 108996.

[17] A. S. Rajawat, S. B. Goyal, P. Bedi, C. Verma, E. I. Ionete, and M. S. Raboaca, "5G-enabled cyber-physical systems for smart transportation using blockchain technology," *Mathematics*, vol. 11, no. 3, p. 679, Jan. 2023.

[18] S. S. Gill et al., "Transformative effects of IoT, blockchain and artificial intelligence on cloud computing: Evolution, vision, trends and open challenges," *Internet Things*, vol. 8, Dec. 2019, Art. no. 100118.

[19] K. Nova, A. Umaamaheshvari, S. S. Jacob, G. Banu, M. S. P. Balaji, and S. Srithar, "Floyd–Warshalls algorithm and modified advanced encryption standard for secured communication in VANET," *Meas., Sensors*, vol. 27, Jun. 2023, Art. no. 100796.

[20] R. Martino and A. Cilardo, "Designing a SHA-256 processor for blockchain-based IoT applications," *Internet Things*, vol. 11, Sep. 2020, Art. no. 100254.

[21] A. W. Kiwelekar, P. Patil, L. D. Netak, and S. U. Waikar, "Blockchain-based security services for fog computing," in *Fog/Edge Computing For Security, Privacy, and Applications*. Cham, Switzerland: Springer, 2021, pp. 271–290.

[22] E. Blasch, R. Xu, Y. Chen, G. Chen, and D. Shen, "Blockchain methods for trusted avionics systems," in *Proc. IEEE Nat. Aerosp. Electron. Conf. (NAECON)*, Jul. 2019, pp. 192–199.

[23] Z. Ma, J. Wang, K. Gai, P. Duan, Y. Zhang, and S. Luo, "Fully homomorphic encryption-based privacy-preserving scheme for cross edge blockchain network," *J. Syst. Archit.*, vol. 134, Jan. 2023, Art. no. 102782.

[24] F. Li, K. Liu, L. Zhang, S. Huang, and Q. Wu, "EHRChain: A blockchain-based EHR system using attribute-based and homomorphic cryptosystem," *IEEE Trans. Services Comput.*, vol. 15, no. 5, pp. 2755–2765, Sep. 2022.

[25] C. Regueiro, I. Seco, S. de Diego, O. Lage, and L. Etxebarria, "Privacy-enhancing distributed protocol for data aggregation based on blockchain and homomorphic encryption," *Inf. Process. Manage.*, vol. 58, no. 6, Nov. 2021, Art. no. 102745.

[26] J. Chen, K. Li, and P. S. Yu, "Privacy-preserving deep learning model for decentralized VANETs using fully homomorphic encryption and blockchain," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 11633–11642, Aug. 2022.

[27] B. Jia, X. Zhang, J. Liu, Y. Zhang, K. Huang, and Y. Liang, "Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in IIoT," *IEEE Trans. Ind. Informat.*, vol. 18, no. 6, pp. 4049–4058, Jun. 2022.

[28] M. Wazid, A. K. Das, and S. Shetty, "An authentication and key management framework for secure and intelligent transportation of Internet of Space Things," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 6, pp. 5242–5257, Jun. 2024.

[29] S. Roy, S. Nandi, R. Maheshwari, S. Shetty, A. K. Das, and P. Lorenz, "Blockchain-based efficient access control with handover policy in IoV-enabled intelligent transportation system," *IEEE Trans. Veh. Technol.*, vol. 73, no. 3, pp. 3009–3024, Mar. 2024.

[30] S. Thapliyal, M. Wazid, D. P. Singh, A. K. Das, and S. H. Islam, "Robust authenticated key agreement protocol for Internet of Vehicles-envisioned intelligent transportation system," *J. Syst. Archit.*, vol. 142, Sep. 2023, Art. no. 102937.

[31] S. Son, D. Kwon, S. Lee, Y. Jeon, A. K. Das, and Y. Park, "Design of secure and lightweight authentication scheme for UAV-enabled intelligent transportation systems using blockchain and PUF," *IEEE Access*, vol. 11, pp. 60240–60253, 2023.

[32] K. Mahmood, J. Ferzund, M. A. Saleem, S. Shamshad, A. K. Das, and Y. Park, "A provably secure mobile user authentication scheme for big data collection in IoT-enabled maritime intelligent transportation system," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 2411–2421, Feb. 2023.

[33] C. Wu, A. Lakhan, and T. M. Gronali, "Microservices architectural based secure and failure aware task assignment schemes in fog-cloud assisted Internet of Things," *Int. J. Intell. Syst.*, vol. 37, no. 11, pp. 8696–8729, 2022.

[34] H. B. Keller, *Numerical Methods for Two-Point Boundary-Value Problems*. New York, NY, USA: Dover, 2018.

[35] M. K. Konopiński, "Shannon diversity index: A call to replace the original Shannon's formula with unbiased estimator in the population genetics studies," *PeerJ*, vol. 8, p. e9391, Jun. 2020.

[36] A. Al Badawi et al., "OpenFHE: Open-source fully homomorphic encryption library," in *Proc. 10th Workshop Encrypted Comput. Appl. Homomorphic Cryptogr.*, 2022, pp. 53–63.

[37] L. Mei, J. Gou, Y. Cai, H. Cao, and Y. Liu, "Realtime mobile bandwidth and handoff predictions in 4G/5G networks," *Comput. Netw.*, vol. 204, Feb. 2022, Art. no. 108736.

Abdullah Lakhan received the Ph.D. degree in computer science from Southeast University, China, in 2020. He is working as an Assistant Professor in cybersecurity. He has a research affiliation with Kristiania University College, Oslo, Norway. He has published over 100 high-quality conference papers and journal articles. His research interests include artificial intelligence, cloud computing, digital health, mobile computing, information security, and intelligent transportation.

Tor-Morten Groenli received the Ph.D. degree in computer science from Brunel University London. He is currently a Professor with the Department of Technology, Kristiania University College, Norway, and the Founding Director of the Research Group for Applied Computer Science and the Mobile Technology Laboratory, Institute of Technology. His primary research interests are mobile computing, the Internet of Things, and software architectures.

Huaming Wu (Senior Member, IEEE) received the B.E. and M.S. degrees in electrical engineering from Harbin Institute of Technology, China, in 2009 and 2011, respectively, and the Ph.D. degree (Hons.) in computer science from Freie Universität Berlin, Germany, in 2015. He is currently a Professor with the Center for Applied Mathematics, Tianjin University, China. His research interests include mobile cloud computing, edge computing, the Internet of Things, deep learning, complex networks, and DNA storage.

Muhammad Younas is working as a Professor of Computer Science School of Engineering, Computing and Mathematics at Oxford Brookes University Headington Campus Oxford. His research interests are web technologies, cloud computing, big data and NoSQL systems, web and database systems cloud and the IoT, and web searching and information retrieval.

George Ghinea (Member, IEEE) received the B.Sc. and B.Sc. (Hons.) degrees in computer science and mathematics and the M.Sc. degree in computer science from the University of the Witwatersrand, Johannesburg, South Africa, in 1993, 1994, and 1996, respectively, and the Ph.D. degree in computer science from the University of Reading, Reading, U.K., in 2000. He is a Professor of Mulsemedia Computing with the Computer Science Department, Brunel University London, Uxbridge, U.K. His research activities lie at the confluence of computer science, media and psychology, focusing on building adaptable cross-layer end-to-end communication systems incorporating user multisensorial and perceptual requirements.