Received: 4 March 2021 | Revised: 28 October 2021 | Accepted: 3 November 2021

DOI: 10.1002/ett.4407

RESEARCH ARTICLE

WILEY

Cyber-physical network architecture for data stream provisioning in complex ecosystems

Kennedy Chinedu Okafor[1] | Michael Chukwudi Ndinechi[2] | Sanjay Misra[3]

[1]Department of Mechatronics Engineering, Federal University of Technology, Owerri, Nigeria

[2]Department of Electrical and Electronic Engineering, Federal University of Technology, Owerri, Nigeria

[3]Department of Computer Science and Communication, Ostfold University College, Halden, Norway

Correspondence
Sanjay Misra, Department of Computer Science and Communication, Ostfold University College, Halden, Norway.
Email: sanjay.misra@hiof.no

Abstract

Intelligent fog cyber-physical social systems (iFog CPSS) is a novel smart city project that uses intrinsic processes to automate microservices such as edge-to-fog or fog-to-cloud monitoring of complex real-time activities. This article presents a dynamic cyber-physical architecture that leverages iFog layers to map location-based services (LBS) on a spine-leaf datacenter clos topology. Individual edge clusters are connected to the edge-fog layer, which communicates with iFog gateways for processing streams' requests. Use-case application of artificial intelligence (AI) in vehicular ad-hoc networks (VANETs) is introduced for data stream provisioning. In the validation study, a secure docker-based iFog CPS experiment is carried out using traffic trace files from C++ modeller. iFog spine-leaf architecture for fog-computing and cloud-computing are compared using two key metrics. For traffic workload utilization, the results show that 83.33% of the traffic workload is utilized at the Fog layer while 16.67% is consumed in the cloud layer. For latency profile, the results indicate that Fog and cloud streams had 20.31% and 77.69%, respectively. In terms of iFog VANET spine-leaf congestion control, three distinct algorithms are studied, namely the proposed linear routing algorithm (LRA), LEACH, and collection tree protocol (CTP). In each case, the resource utilization for VANET gave 34.45%, 32.18%, and 33.37%, respectively. Latency response gave 11.76%, 82.35%, and 5.89%, respectively. Also, the throughput scenario offered 19.61%, 39.22%, and 41.17%, respectively. Consequently, the iFog scenario offers satisfactory LBS provisioning for VANETs clusters.

1 | INTRODUCTION

Intelligent Fog computing for smart cities has recently attracted several research interests in today's world.[1] The motivation is on cloud elasticity in complex ecosystems such as Smart City VANETs and other edge applications. Such streamlined massive transition into more organized societies is often called smart society. Artificial intelligence (AI) adoption within cloud, fog, and edge computing paradigms has made intelligent societies a reality. Some recent contributions in cyber-physical systems (CPS) include vehicular network compressive sensing (CS) technology,[2] IoT smart disaster management,[3] smart city vehicular edge server design,[4] and energy-efficient workload distribution,[5] among others. Interestingly, these are all provisioned from the dedicated cloud deployments. A new alternative deployment could happen at the Fog layer using software-defined approaches with machine learning algorithms.

Recently, the scientific community is now seeking to leverage the new concept of CPS to solve complex problems in organic ecosystems. These are mostly coordinated networked control distributed systems (CNCDS) with active feedback algorithms. Usually, these systems' components are distributed in a spatial form and linked via high-speed communication networks. Typical examples include cyber-driven manufacturing, smart grid, Industry 4.0, and recently smart societies. These smart ecosystems make use of automated processes for service provisioning. But the aggregation of intelligent systems that use established values to drive standards for enhanced living is referred to as smart society (SS). In such systems, multichannel telecommunications architectures are required to achieve seamless transactions.[6] Japan is the first society to introduce a new coinage known as Society 5.0.[7] This societal model constructs a sustainable environment for human existence while taking cognizance of security, well-being leveraging CPS. Such society includes wider societal participation using digitization technological supports that benchmark trustworthiness indices.[8] The evolution of CPS has tremendously moved society into a world of intelligent interactions between humans and objects in the physical world.

Moreover, a value-driven society creates systems and processes that can allow citizens to leverage analytics for decision-making. This leads to societal stability, well-being, environmental sustainability, social standards, and good governance.[1] For instance, the use of smart car parking systems (SCPS)[8] involving vehicular networking (VN) with convoluted AI communication platforms can be used to address transportation problems in urban centers.[9]

With the increasing quest for urbanization, the demand for intelligent and sustainable environmental ecosystems that provide citizens with a satisfactory quality of life remains indispensable. This involves unifying technologies that interact with governments, citizens, and processes to support entities such as smart energy, smart mobility, smart environment, smart living, and smart governance, among others. For most African nations, the advancement towards a better society will involve massive leverage on intelligent technologies and smart electronics involving the Internet of Things (IoT),[10] mobile big data, and fog computing analytics that depend on real-time processing.[11-13]

Therefore, to achieve smart urbanization, a robust system architecture that will guarantee data's trustworthiness is needed for high-bandwidth delivery, low-latency diffusion, and zero-blocking backend server connections. Hence, reliability, availability, and security are key in VANET CPS clos-topology. The summarized contributions of this research include:

- i To derive iFog game optimization model for scalability and QoS dynamic provisioning.
- ii To establish use case scenarios relying on iFog layers as optimization constraints.
- iii To show feasible integration for datastream workload platform
- iv To carry out a spine-leaf evaluation based on real-world C++ trace files needed for the validation of the iFog framework.

The article is organized in the following section. The related work is summarized in Section 2. The proposed model architecture is given in Section 3. Section 4 presents the proposed cyber-physical implementation scenario. Section 5 presents the experimental simulations. Section 7 presents the use case scenario deployment framework. Section 8 concludes the work with future directions.

2 | RELATED WORKS

There have been increasing research efforts recently on disruptive smart societies. The primary issue is identifying a robust architecture that is well characterized to support massive computational workloads using AI standards and protocols. This section presents closely related research efforts in this regard using a systematic literature review for quantitative analysis. The idea of a smart city architectural model that explores semantic computing to achieve city urbanization vision has been ongoing.[14] In such design, a conceptual design with group management procedures has been documented.[15] One such conceptualization is an open-source smart city implementation for smart transportation, smart home automation, smart unmanned aerial vehicles, and smart energy grid, among others.[16] These leverage machine-to-machine (M2M) communication while considering an ethically driven smart society.[17] Most of the smart city deployments use smart web applications for industrial process maintenance operations.[18] Other complex ecosystems explore smart society paradigms such as in mobile application and smart city orientation,[19] smart garbage collection in urban settings,[20] smart home infrastructure,[21] smart grid architecture, and automation.[22,23] Deeper contributions in cyber-physical social systems (CPSS) have been studied. For instance, the work[24] focused on advanced Industry 5.0, which relies on CPSS.

In Reference 25, AI is applied for evidence-based policy in Society 5.0. In Reference 26, the authors presented a Fog-supported smart city model for the management of IoT-powered applications. The authors in Reference 27 looked at a secured but dependable multilayered demand side management model for smart social society using IIoT-based big data analytics. The computational requirements for their data streams need further investigation. A summarized systematic literature review (SLR) has been highlighted in Table 1 to offer more insights on CPS domains.

In References 40,41, CPS powered by AI applications is well highlighted. From the findings in Table 1, the field domain of AI-driven CPS will continue to grow significantly. This is because the fusion of AI in the modern CPS will continue to be facilitated by AI services providers. These providers support flexible integration even with endpoints supervisory control and data acquisition (SCADA) servers that host ubiquitous computing services. Another growth factor is that the CPS powered by big data, IoT, is active as emerging technologies solve diverse societal problems. However, the technology protocols and standards in CPS are yet to be fully established since the legacy smart city model is hindered by inadequate data stream benchmarking.[42] In this regard, quality of service (QoS) provisioning in stream computing is a major concern.

The major research gap in most works is that little efforts have been made in AI optimization schemes needed for scalability and QoS dynamic provisioning in smart city vehicular data stream provisioning. Also, there are no layered use-case formulations for optimization constraints in smart city VANETs. The layered algorithms for workload computations are not obvious in the literature. Finally, the use of spine-leaf design implementation is yet to be applied in CPSS.

Motivated by this consideration, a dynamic cyber-physical architecture that uses iFog layers to map services for analytics is implemented. Spine-leaf datacenter is used to determine the efficiency of the system under stream traffic workloads. The key feature of iFog CPSS is that CPS-Fog nodes support end-to-end data streams movement in the era of Industry 4.0. With an AI algorithm involving Kubeflow pipelines for scalable replication in disaster incidence, the architecture in Figure 1 is preserved while offering better workload data stream processing outside the cloud without elasticity overhead. The implication is that critical QoS metrics will be largely optimized at the Fog layer, thereby offering more efficient data streams provisioning for the entire system.

TABLE 1 Cyber-physical network efforts for Society 5.0 AI environment

Reference	AI application	Technology enabler	Network type	Protocols/standards	Modules
28	Non-AI	IoT cloud CPS	Peer-to-Peer network CSMA/CA,	IEEE 802.15.4	Distributed cloud
29	AI-enabled	CPS	Not clear	All networks	Smart cities
30	Agent systems	CPS	IT/OT convergence	MAS-based control system	SCADA
31	AI enabled	Industrial control system (ICS) and Industry 4.0	Distributed ICS testbed	Undefined	Industry 4.0
32	AI enabled	CPS	Not clear	Undefined	Intelligent vehicles future challenge (IVFC)
33	AI-enabled	CPS	Multicore fuzzy logic system (FLS)	Undefined	Automated guided vehicles (AGV)
34	Non-AI	CPS	Not clear	Undefined	SmartPM
35	AI-enabled	CPS	Bidirectional long short term memory (LSTM)	Undefined	SmartGrid hybrid tracked vehicle (HTV)
36	AI-enabled	CPS	HPC	Undefined	Industry 4.0.
37	AI-enabled	CPS	Service-oriented architecture	Undefined	Industry 4.0.
38	AI-enabled	CPS	AIS-fuzzy.	Undefined	AIS-fuzzy multirobot
39	AI-enabled	Undefined	Industrial AI.	Undefined	High-speed railway (HSR) transportation

3 | THE PROPOSED ARCHITECTURE

3.1 | CPSS iFog computing architecture

This section discussed the proposed iFog CPSS architecture in Figure 1. This illustrates the intelligent fog (iFog) system for service provisioning in a smart society running various services.

The iFog computing architecture has its application services and components running on both cloud and endpoint devices via its smart gateways. A distinctive six-layer architecture for CPSS is described below.

3.1.1 | Layer 1: Edge computing layer

As shown in Figure 1, we introduced the iFog physical/virtualization layer that houses the entities (or nodes $i \ldots \ldots j$) connected to the internet cloud I_c while generating I/O data streams for analytics. This is achieved with RDS3115 metal gear servo motor, Arduino Uno, RDM6300 RFID module) shown in Figure 2. Also, end-user devices like Telos-B sensor

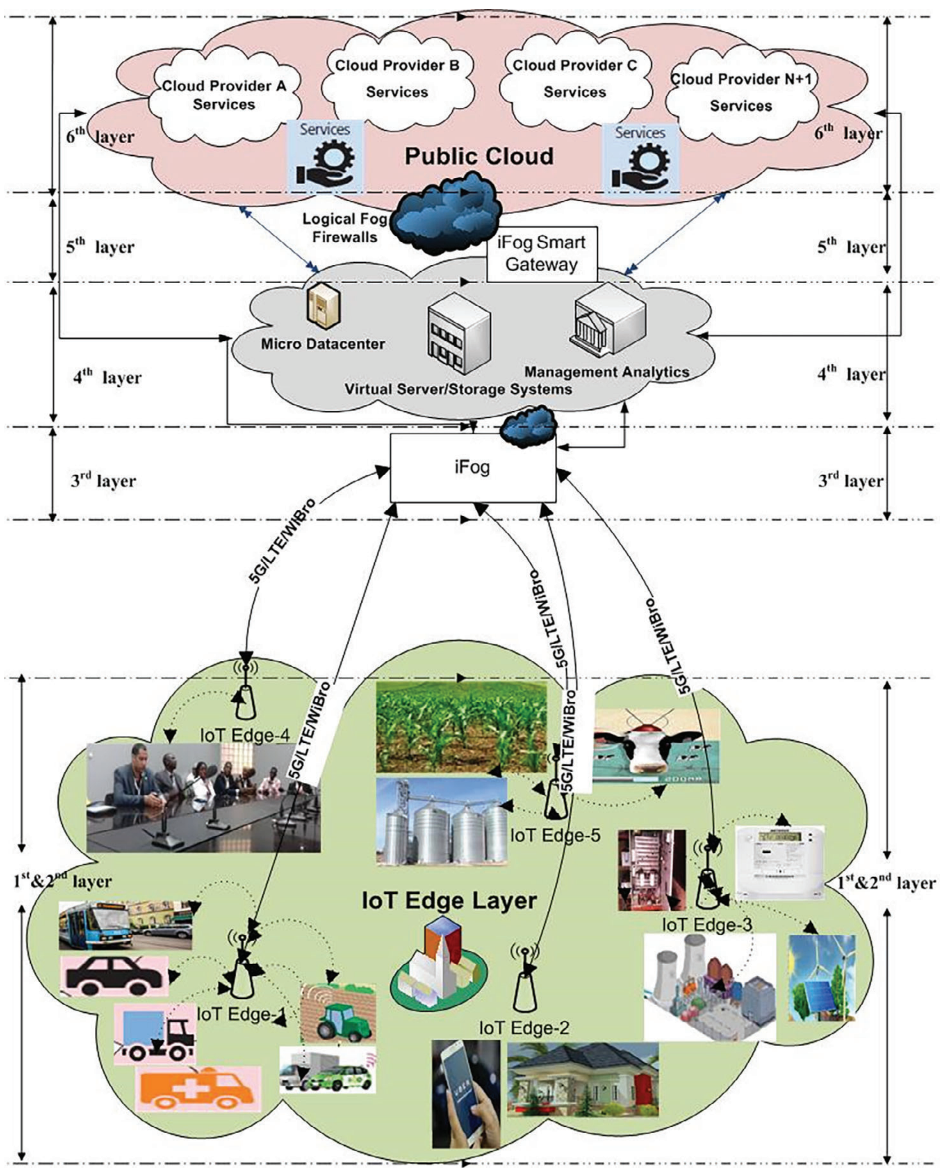

FIGURE 1 Dynamic fog computing architecture for iFog CPSS (smart city)

integration are applied at this layer.[43] Other useful sensors are found in References 44,45. These nodes are handled based on their service requirements and characteristics. IoT networked devices, local computing, ubiquitous dial-in accessibilities with limited processing and storage are found in this layer. In this work, the proof has been implemented in Figure 2.

3.1.2 | Layer 2: Access point monitoring (APM) layer

Figure 3 is the access point monitoring layer, where the edge nodes' events and activities are transferred to the iFog gateway using the MK1000 microcontroller device. This layer determines how the service set identifier is distributed to the upper layers.

At this layer, let $A = \left(A_{1\,\ldots,}A_n\right)$ and $B = \left(B_{1\,\ldots,}B_n\right)$ depict the various edge subsets of a finite set S. Then the APM layer monitoring equation is given by Equation (1)

$$S = \left| \sum_{i,j}^{\infty} \bigcup_{i,j} \left\{ i : B_i \cap \left(A_j\right) \neq 0 \right\} \right| \geq |i, J| \; for\ all\ i, j \setminus in \tag{1}$$

where A and B represent two distinct edge devices in CPS and its intersections depicted by S. O in other edge devices can be introduced within S.

3.1.3 | Layer 3: Preprocessing layer

At layer 3, the iFog preprocessing layer primarily handles I/O data streams in VANET cooperative awareness messages (CAMs) and multihop message dispatch. iFog software-defined controller[40] (iFog-SDC) uses RESTFUL API to orchestrate the VANET data and control plane via roadside units (RSU), as shown in Figure 4. Three major components are briefly discussed.

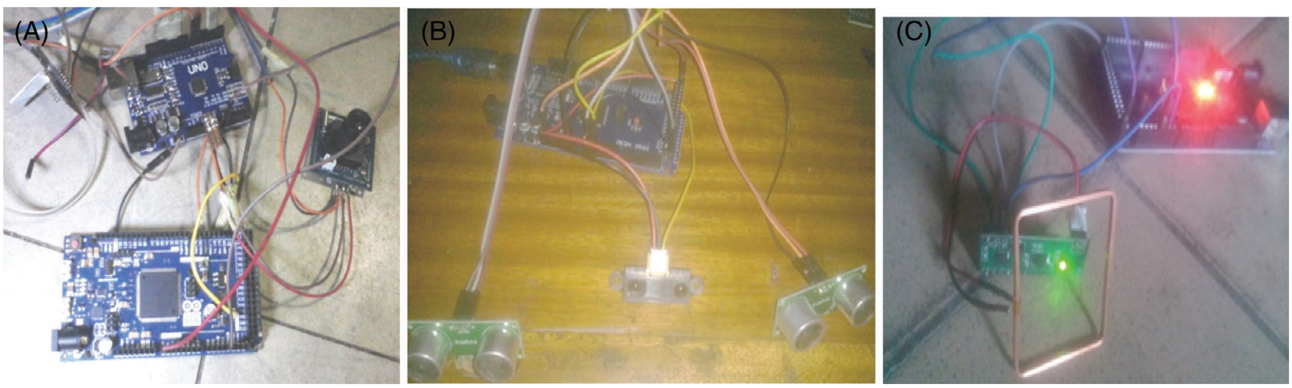

FIGURE 2 (A) Edge layered camera module with arduino board mix, (B) edge ultrasonic distance sensor (GH-311RT), and (C) edge RFID Module communication with arduino board(Source: Authors Testbed)

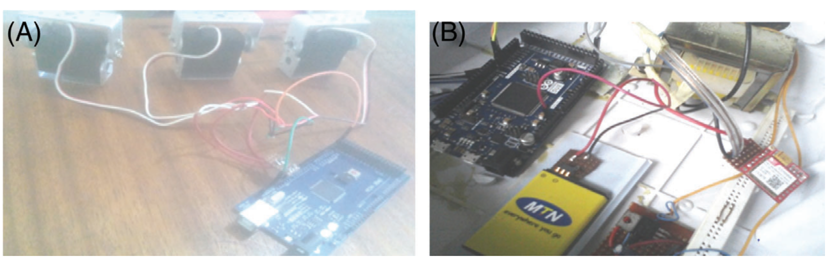

FIGURE 3 (A,B) Layer 2 APM with a service provider network (Source: Authors Testbed)

Management: The iFog SDN executes the filtration of records from VANET and transforms them using its analytic service nodes. Yang-based network configuration protocol (NETCONF) is used as a network management protocol for statistical monitoring of the movements. The base protocol includes the following protocol operations: get, get-config, edit-config, copy-config, delete-config, lock, unlock, close-session, and kill the session.

Filtration: Stateful layer-3 firewall s_f is used to protect the cloud stack availability zones from layer 3. Through stateful firewall filtration, both the secondary storage and pod servers are preserved. The IOS-XE YANG version 16.5 offers robust filtration supports.

Transformation: VANET unstructured datasets V_{ud} have streams-data sampled at irregular intervals. With normalization N_V, VANET datasets from both structured and unstructured components are reorganized using the self-organizing map (SOM) for IBM Cloud analytics.

3.1.4 | Layer 4: Storage buffer

The iFog storage layer 4, serves as the temporal input-output (I/O) data-streams storage buffer. Virtual machine footprints, VANET data streams, and snapshots. The cloud Infrastructure as a Service (IaaS) is used to store huge datasets for a longer duration compared with the Fog due to its vast storage capability. At this layer, let $N = (G, c, s, t)$ be the storage graph flow network comprising micro datacenter M_d, virtual server/storage systems VS_{SS}, and management analytics M_a.

3.1.5 | Layer 5: Security layer

As VANET data streams are pushed from the iFog SDN layer into the cloud, stateful logical firewalls are located at this layer. However, AI-based firewalls are considered against the signature-based firewalls that seem to be reactive, resource-intensive, and vulnerable.

3.1.6 | Layer 6: Converged cloud layer

This layer depicts the transport layer evoked when iFog is ready for transportation into the cloud or the edge. Layer 6 converged CloudStack is designed to work with all standards-compliant iSCSI and NFS servers supported by hypervisors

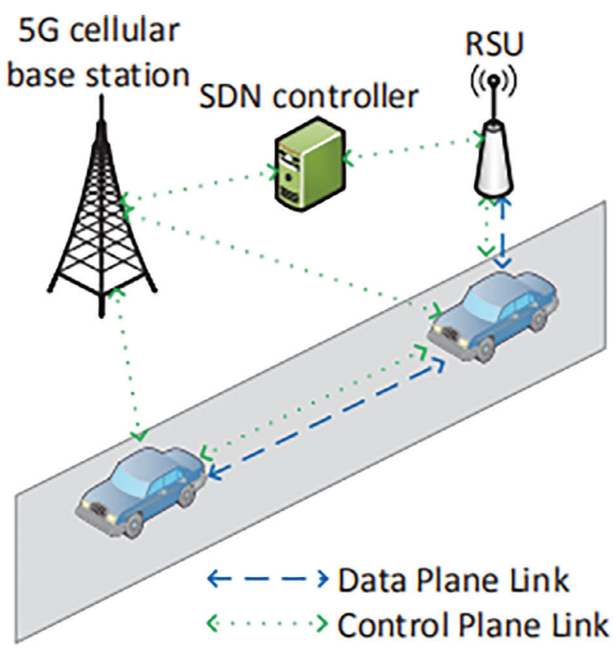

FIGURE 4 SDN VANET Control and data planes with the 5G cellular network

based on dynamic auto-scaling. At this layer, resource orchestration is considered (such as onboard, provision, service manage, CPU, storage, and network), workload orchestration (ie, workload ware placement, optimization, and operation), service orchestration (ie, VANET lifecycle management). Furthermore, this layer has unlimited storage and processing capabilities and supports high-performance computing, availability, and high latency. In practice, the converged cloud layer uses dynamic auto-scale to manage systems resources, including storage. Based on On-demand service provisioning, the unlimited capacity can now be auto-scaled. This is the idea behind cloud elasticity. An optimal computation programming model for layer 6 is given by Equation (2)

$$Maximize \sum_{i,j=1}^{\infty} \prod QoS(Q) * C(x),$$

$$Subject\ to\ Cg_i(x) \leq gb_i, i = 1, \ldots \ \ldots \ \ldots ., \infty,$$

$$Ch_j(x) = hb_i, j = 1, \ldots \ \ldots \ \ldots ., \infty, x \leq 0,$$

where the objective function C is a function of complex variable x, and the constraint functions Cg_i and Ch_j are general Cloud dependencies or functions of the variable x.

The major concern with the iFog application is in designing a computational model that will support QoS workload processing. In this case, the latency requirement for high-quality streaming to mobile nodes and vehicular movement from proxies and access points must be supported on the network.

In this article, the proposed CPS iFog uses AI dynamic adjustments to guarantee fair resource allocation and QoS for better bandwidth utilization. This is the major function of the robust Spine-leaf architecture applied in Section 5. It supports scalability and redundant replications. Also, various use case applications of AI in the vehicular ad-hoc network (VANET) will be discussed.

3.2 | CPSS iFog cloud datacenter AI network model

This section highlights classical perspectives on CPS efforts for the Society 5.0 environment using AI and its related concepts. The primary features of AI-driven CPSS (iFog architecture) discussed include scalability and the presented redundant replication of traffic workloads (which normally affect QoS metrics). In this case, VANET is the reference CPS discussed throughout this research.

3.2.1 | Artificial intelligence infusion

First, the scalability model required for throughput and latency metrics is managed by the Kubeflow-pipeline Algorithm I. The proposed CPS *iFog* engages AI-powered SDN to manage the VANET nodes and servers in the CPS ecosystem. The iFog layer extracts I/O data streams, thinks, processes, and accurately executes data streams based on prior knowledge.

Using the spine-leaf architecture in Figure 1, massive deployment of edge devices/nodes is considered while using optimization controls to achieve the best state processes. For smarter functionality, the entire layered spectrum in Figure 1 can leverage AI supports. Examples of such include adaptive natural language processing, middleware logic, automated operations, computational machine learning orchestration, and focused game models.[41] In terms of security, applications in the system can give room to geometric complexity attacks (such as ransomware), which can weaken the network in the absence of secured AI algorithms.

Stateful firewalls offer smarter supports for countering the increasing threats presented by malicious attackers. In this case, the use of an AI algorithm can address cybersecurity concerns in Figure 1 by determining both the strengths and weaknesses of the various layers. For the CPS iFog ecosystem to work, an AI integration algorithm addresses the communication, computation, and monitoring functionalities. This enables data collection using edge layered sensor networks (ELSN) and embedded systems for an environmental response via software components and actuators. The use of Kubeflow-pipeline to learn the data streams is deemed significant in the use case scenario discussed in Section 6.

3.2.2 | Data mining AI engine

CPS iFog ecosystem can employ various AI models such as regression, classification with artificial neural network (ANN), and Bayesian network and clustering (K-means). However, this work adopted neural network clustering as the LBS knowledge discovery framework. This is used to determine vehicle fleets' effective tracking strategy in peace mass logistics (PMT) company, Nigeria. Cluster analysis data mining is used to detect vehicle halting and to stay around someplace or similarity pattern set. Sequential characteristics wherein every record are maintained or regular time basis is observed. This is achieved with Cluster analysis, which detects spatial-temporal datasets. This can be used to identify the reasons why accidents happen on traffic highways. Data preparation in DM includes data selection, filtering, transformation, creation, integration, and formatting. The fourth phase is modeling. It involves the use of analytical methods (algorithms). Such data are used in the LBS clustering for vehicular arrivals.

3.3 | AI use case application

3.3.1 | Location-based service data mining knowledge discovery

In this section, an AI data mining drive test application is employed in the production LBS domains for CPS iFog independent random variables. The application is integrated with a global positioning system (GPS), and this is enabled on VANETs. This forms a CPS location-based mining dataset leveraging knowledge discovery methodology. Peace mass transport company is adopted after wide consultations with various transport haulage logistics organizations in Nigeria. In LBS, data mining is a process that is used to identify hidden, unexpected patterns or relationships in large quantities of datasets. Finding useful patterns in datasets of a case study represents data mining or knowledge extraction in CPS deployment. This use case scenario case study is applied in haulage data description testbed and peace mass transit software. However, the CPS solution is used as a trajectory database repository. Geospatial data analysis is presented in the use case. The AI engine works with the clustering spatial-temporal data system (CSTDS) principle shown in Figure 5.

3.3.2 | CPS- LBS architecture

Figure 5 depicts the CPS use case of VANET architecture that runs CPS iFog AI algorithms. This comprises four or more subnet clusters of the vehicle to vehicle (V2V) model. The component elements of each VANET architecture include OnBoard Units (OBU) and road side unit (RSU). OBU is a radio device installed in the vehicle, in turn, that RSU places along the road and works as a router between vehicles. OBUs integrate with the onboard systems using their available computing resources, an antenna, and an output dashboard. Included in the firmware and hardware units are the VANET communication interfaces. This supports modular communication via the redundant TCP/IP router in a smart city.

3.3.3 | CPS VANET data stream exchange

As shown in Figure 5, an identified method of transferring data streams within CPS-VANETs is highlighted. In this case, the trajectory services use TCP/UDP + IPv6 link bundle. But for road safety records, the statistical data collection on the state of traffic flows is achieved with the AI geospatial application. AI collection tree protocol (CTP)[46] is employed and implemented in the application. The benefits include reliability, efficiency, and hardware independence, especially from the edge sensors running on the CPS iFog model.

3.3.4 | CPS VANET traffic database systems

As shown in Figure 6, trajectory data clustering and mining (TDC-DM) servers between various VANET clusters are introduced. The VANET TDC-DM interchange servers are equipped with complex AI databases with signal timing repository (STR), traffic information management system (TIMS), and deterministic data warehousing for haulage logistics

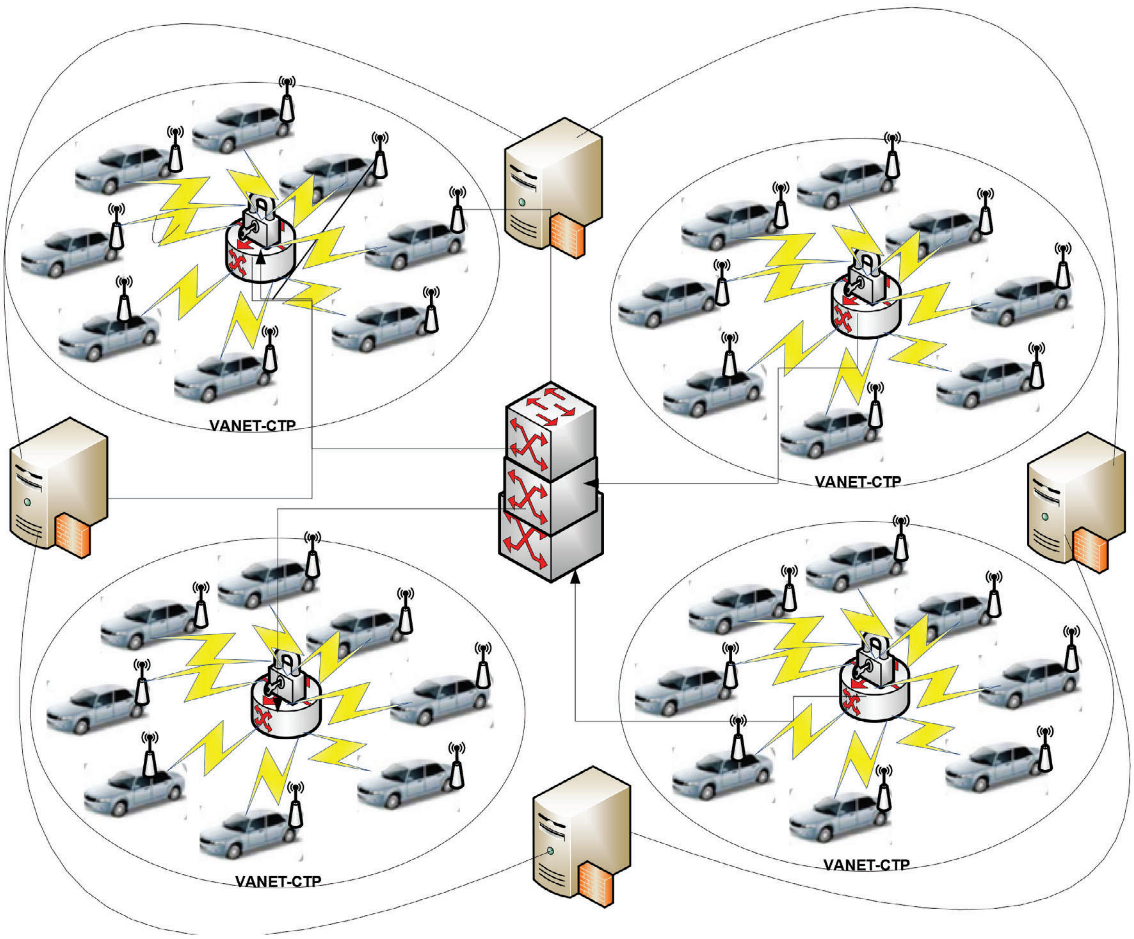

FIGURE 5 CPS use case scenario—VANET architecture

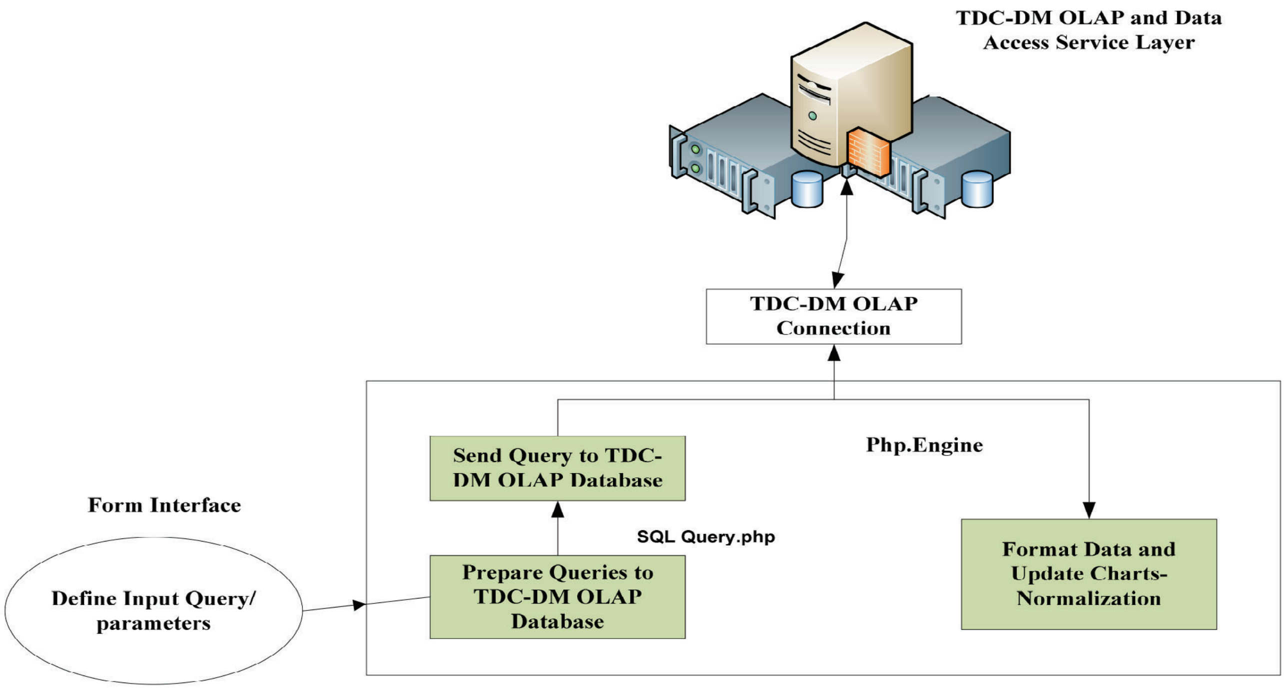

FIGURE 6 VANET TDC-DM query workflow framework

decision support others. The TDC-DM warehouse serves at the backend that processes real-time data streams. The TDC-DM for V2V logistics decision support includes advanced traffic controllers with video cameras, sensors, data aggregation, and intersection readers within square block areas. The system deployment is expanded to road/street_A, road/street, road/street, and so forth.

The VANET TDC-DM architecture uses TIMS to extract traffic data within the data warehouse analytical processor. The TIMS data includes *N*-Automatic traffic recorder (ATR) counts, *R*- road *id* counts, *K* segments/intersections, *T* turning movement and vehicle classification counts, and so forth. An integrated data mining system (DMS) makes for efficient real-time control and monitoring of haulage logistics, as shown in Figure 6.

3.3.5 | CPS VANET trajectory traffic data mining

From Figure 7, the various VANET trajectory traffic data mining components include:

TDC-DM database
In the VANET CPS, the online analytical processing (OLAP) cube is configured for a multidimensional array of datasets. It is a server-based technique depicted in Figure 4 for analyzing the VANET data stream necessary for trajectory insights. The OLAP cube depicts a multidimensional dataset optimizer. The servers carry out logical data processing, data manipulation, and deep geospatial analysis. The TIMS and STR database system are in this category. Normalization is carried using the python pandas tool. This distributes elements to fit the needs of the analysis required in real-time traffic control, as shown in Figure 5. As expected, the OLAP cube is enabled to run routinely for additional analytics. Various data mining capabilities are done via the embedded database layer with built-in database libraries, as shown in Figure 7.

TDC-DM input layer
In the TDC-DM scenario, the data comes from the application processes and input systems of VANET. However, in traffic engineering, data are collected on the motion. These data cover vehicles, tricycles, and pedestrian volumes, and travel times. These datasets are harvested into a signature database via automation. In context, management of data mining at the data inputs needs the use of smart devices such as smartphones or tablets. These devices are not just devices for personal use but are used as data input devices.

Furthermore, mobile devices offer inexpensive and versatile capabilities for application integration. For instance, the edge nodes support Wi-Fi connection, camera, and GPS. TDC-DM uses a data collection server to house the data can containing GPS information, accurate data.

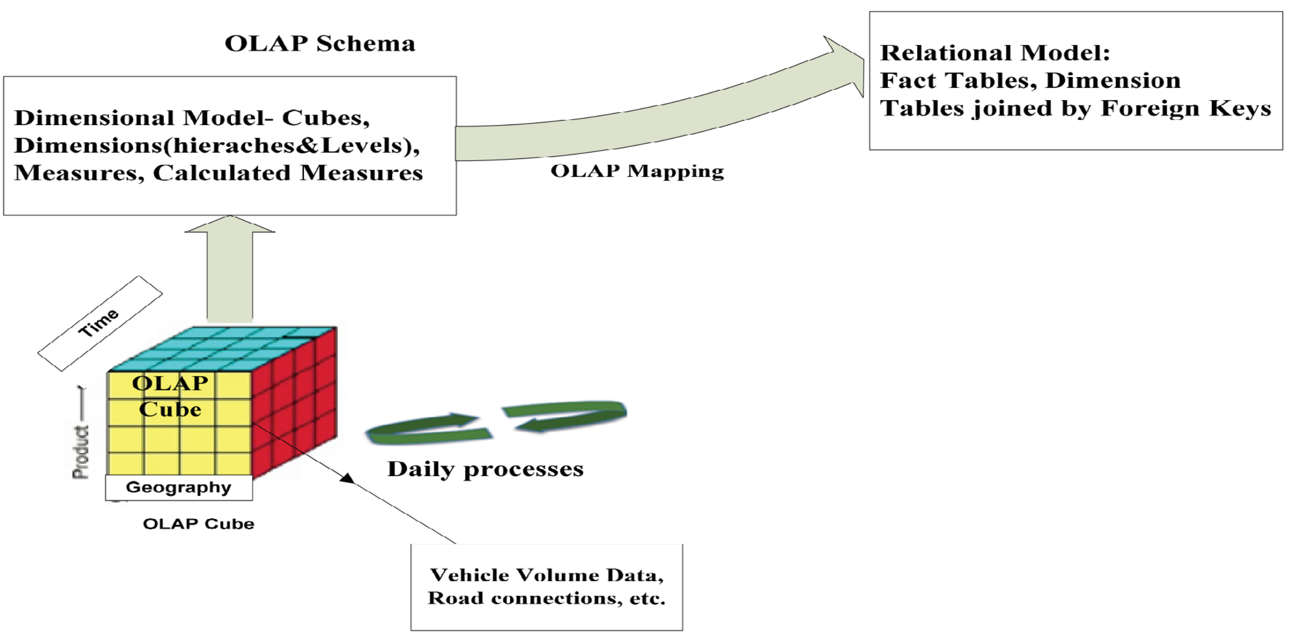

FIGURE 7 VANET Data schema for TDC-DM

TDC-DM OLAP and data access layer

Figure 8 shows the trajectory data mining for V2V logistics decision support using a hand-held device and smartphone. As depicted in figure, this is the especially important aspect of this project since this supports supervised data mining using neural network clustering (SDM-NNC). At the implementation level, the online line analytical processing (OLAP) is characterized by SDM-NNC. The OLAP serves as a basic tool that helps in traffic analysis such as multidimensional datasets, such as time series, and trend analysis of extensive periods. The TDC-DM database stores the big datasets, while the OLAP handles database records in multiple formats and dimensions. In context, an estimated huge dataset requires analysis using customizable dimensions with an OLAP toolbox. The other layers include the TDC-DM services layer and the TDC-DM analysis layer, which captures statistical analysis for presenting time trends and summary of travel times or speeds along with road segments. Another layer known as the TDC-DM presentation layer provides the actual and tangible format of the collated data for the users/analysts. These use view manipulates and processes the datasets finally delivered through this layer. This can be integrated into advanced platforms providing visualization gathered from the processed data.

3.4 | CPS data mining application and integration

In Figure 8, the GPS application tracker provides a platform for data collection, normalization, and visualization. For the CPS iFog scenario, this allows data previews for intelligent pattern tracking and checks. Analytic layers are activated as soon as new datasets are uploaded. For instance, upon uploading the present state data in the database via the CPS iFog layers, the internal database modules for analysis performance, like service SLAs, QoS, and demand-pull, are instantiated. The GPS data from the case study environment (peace mass transit) needs lengthy processing times. As such, data mining with automated algorithms may be scheduled for peak and nonpeak hour periods depending. With TDC-DM in place, complex datasets can be homogenously classified for understanding different formats and data structures.

The VANET data mining design has limitations and challenges with the existing DM schemes studied in peace mass deployment. Most architecture runs on networks that suffer from outages due to the absence of fault tolerance on the network designs. Also, their data mining/KDD process is largely an open-loop system, hence nonsupervisory trajectory DM. Figure 9 shows the formulated design for data mining and the KDD process in TDC-DM. In CPS iFog, the scheme can predict future trends and behavior while allowing instantaneous proactive, knowledge-driven decisions in smart cities. It reduces complex datasets into a simplified model that is useful in predictive analytics.

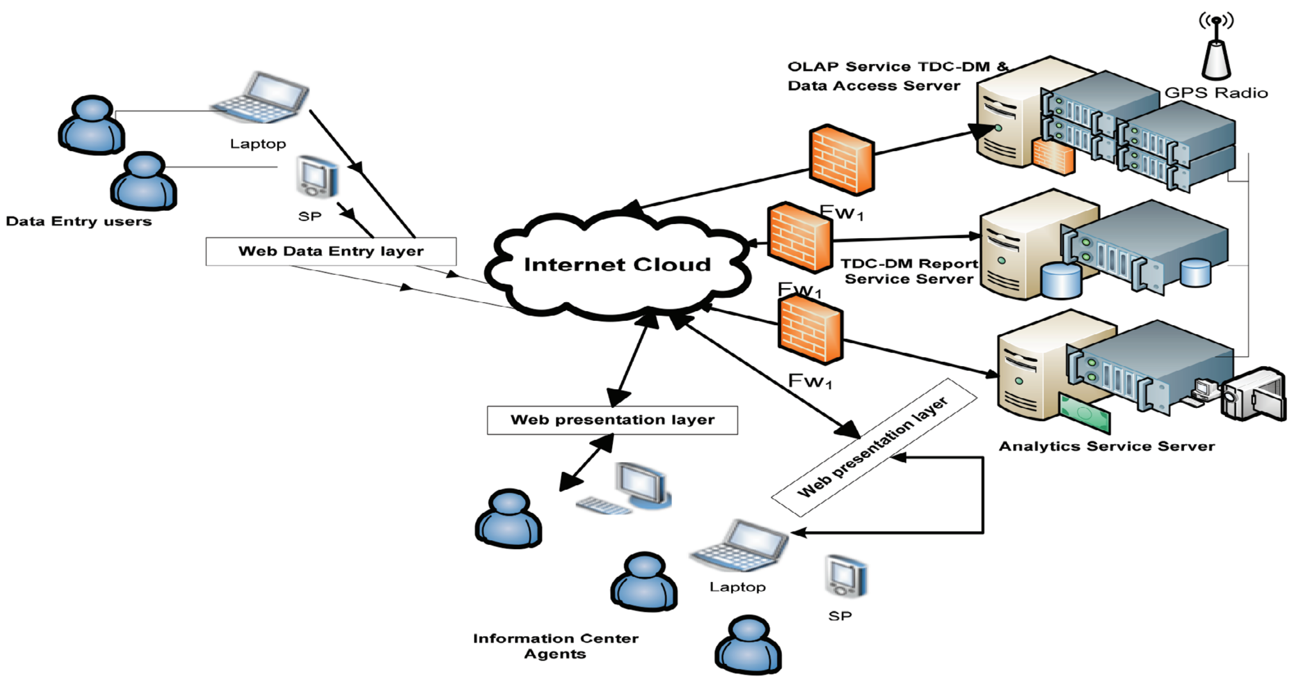

FIGURE 8 CPS VANET trajectory data mining architecture for V2V logistics decision support

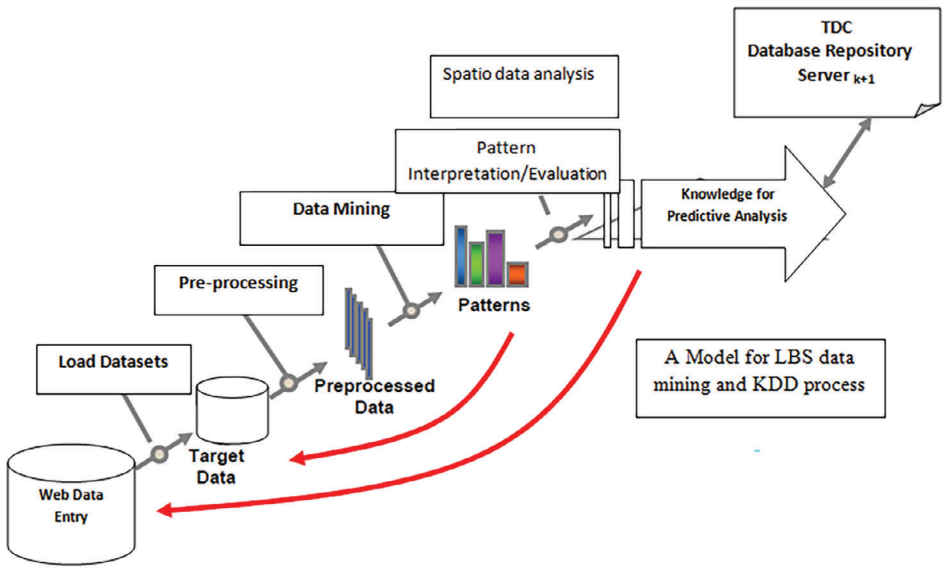

FIGURE 9 CPS iFog LBS data mining and KDD process

CPS iFog service application using VANET in smart city ecosystem needs to address QoS patterns from clustered VANETS. In this case, a smart, collaborative optimization engine is applied to VANETs such as UBER services that offer shared-ride problems. The merit of this work include AI optimization of CPS VANETs for service efficiency, congestion management, and QoS provisioning for complex CPS applications.

3.5 | Mathematical modeling and optimization

3.5.1 | Collaborative optimization

In this section, we shall introduce the collaborative optimization architecture for CPS in Figure 1. This will break the utilization problem space at various layers needed for efficient system architecture. The ith subproblem has the form:

$$\underset{x^{(i)}}{Min}\, f_i\left(x^{(i)}, y^{(i)}\right)$$

$$\textit{Subject to}\ \left[x^{(i)}, y^{(i)}\right] \in \varkappa_i, \tag{2}$$

with $x^{(i)}$ containing a subset of the design variables x and response variables $y^{(i)}$. The constraint ensures that the solution satisfies VANET-specific constraints. Interdisciplinary compatibility requires that the VANET global variables $x_g^{(i)}$ and $y_g^{(i)}$ agree between all entities. We define a set of coupling variables A_g that includes variables corresponding to all design and response variables that are global in at least one subproblem. The agreement is enforced by constraining each $x_g^{(i)}$ and $y_g^{(i)}$ to match its corresponding coupling variables:

$$x_g^{(i)} = A_g\left[x_g^{(i)}\right] \text{ and } y_g^{(i)} = A_g\left[y_g^{(i)}\right], \tag{3}$$

where $A_g\left[x_g^{(i)}\right]$ and $A_g\left[y_g^{(i)}\right]$ are the coupling variables corresponding to the global.

design and response variables in the ith discipline. This constraint is enforced using the subproblem objective function in the CPS model given by Equation (4):

$$f_i = \left\|x_g^{(i)} - A_g\left[y_g^{(i)}\right]\right\|_2^2 + \left\|y_g^{(i)} - A_g\left[y_g^{(i)}\right]\right\|_2^2. \tag{4}$$

Each subproblem thus seeks feasible solutions that minimally deviate from the coupling variables. The subproblems are managed by a system-level optimizer that is responsible for optimizing the coupling variables A_g to minimize the objective function.

3.6 | CPS metrics models

Having discussed the cooperative optimization derivatives in Equation (4), let us establish equations that demonstrate iFog process metrics generation in terms of time, rate of computation/utilization, streams interchange rate, delivery packet rate considering IoT domain, fog, and cloud.

By introducing Equation (1) at the fog, let cloud service rate (CSR) and fog datastream (FD) be considered in the utilization metric. Also, let datastream interarrival rates (DIR) and fog datastream inclusion (FDI) be considered also

$$\text{Let } R_f = \text{Utilization Response from Fog} = \sum (CSR + FD + S), \tag{5}$$

$$\text{let } R_d = \text{utilization delay} = \sum (DIR + FDI), \tag{6}$$

$$\oiint \left(\int_0^\infty \sum_{i=1}^n R_c \right) dt = \int \left(\int_0^\infty \sum_{i=1}^n C_{\{Cc\}} + \int_0^\infty \sum_{i=1}^n f_{PD} \right) dt, \tag{7}$$

$$\oiint \left(\int_0^\infty \sum_{i=1}^n R_d \right) dt = \int \left(\int_0^\infty \sum_{i=1}^n C_d + \int_0^\infty \sum_{i=1}^n P_i \right) dt. \tag{8}$$

Equation (7) and (8) remain very significant. Cloud utilization active state is described by $\boldsymbol{R_c}$. $\boldsymbol{C_{cc}}$ explains the cloud scheduling processes from the iFog layer, $\boldsymbol{F_{pd}}$ describes the datastream delivery from the fog layer while $\boldsymbol{R_d}$ explains the latency profile for all the processes. Since the transactional processes may execute at various timestamps, we used $\boldsymbol{C_d}$ as the new latency delay for all run-time processes. $\boldsymbol{P_i}$ describes the interarrival latency delay on schedule. These equations are applied for services running on both the fog and cloud layers and verified by the experimental results.

3.7 | CPS Kubeflow-pipeline algorithm

Algorithm I show the AI engine responsible for full-stack integration involving the end-users as social actors. Lines 1–9 explain how the VANET uses Kubeflow-pipeline to provision data streams from the edge to the cloud using critical test conditions. This is a vital smart city CNCDS. Pipeline flow allows users to generate data streams and profiles in Kubeflow for orchestration and analytics.

For the social aspect, Algorithm I uses its ranking function in Lines 10–15 to identify, display and distribute VANET content to end-users feeds with minimum latency. New update information is released each time a query is made in Line 15. The social relevance here is to show edge users priority information that matters most and how it can equally manipulate individual VANETs. With collaborative optimization, Connection and engagement in Line 16-29 are then used to complete the VANET ranking of its contents to end users' devices.

Algorithm 1. iFog AI Sandbox // Kubeflow-Pipelines

1:Input: CPSS authentication requests from SmartCity Edge_Devices.
2:Initialization: SmartCity_Edge_Devices = NULL, Edge_Device = 0; Kubeflow-Pipelines = 0.
Let edge users $u(x) = \{u1, u2, \dots, n+1\}$ denote the indexing string at the Edge of the CPSS.
//Accumulator stream images A*si* can be computed in the Kubernetes gallery *Kg*.
3:Output: Either connect devices or reject devices & Auto-scale = iFog cloud datacenter.
4:Procedure: Edge-to-cloud processes ().
Setup Kubeflow-pipelines using IBM cloud TensorFlow extended (TFX) //engine clusters.
5: Call cloud storage space bucket ().
6: Deploy Kubeflow-pipelines for 15 minutes.

7: Let i represent the edge input devices/vehicles $\{1 \ldots \ldots \ldots n+1\}$ denoting polynomial edge.
vehicles derived from clusters Xj, and $\{ki = 1 \ldots \ldots \ldots \ldots n+1)$.
8: Set aggregators for all centroids to zero.
9: **While** (Kubeflow-Pipelines j) **do //** unsupervised ML for running, monitoring, auditing,
//managing ML pipelines on IBM Kubernetes.
10: Take a snapshot of edge devices ()
Train VANET for cluster analysis ().
11: Set Cloud_SmartcityEdge_Devices on incremental states ().
12: End while.
13: For each Edge clusters **do.**
14: Find edge device with the closest centroid// Rank, sort & display available VANETS.
15: Trigger alert for an identified device// Advertise new update information.
16: increment accumulator for edge device K, $K_1 + K_2 \ldots \ldots \ldots \ldots .K_{1+n.}$
17: End if
18: End for
19: If all the edge device complete data stream cycles, **then**
20: Engage pipeline workflow //Call Results.Kubeflow-Pipelines
//Let Kubeflow Pipelines utilize critical dependencies && map pipeline's workflow
//as a directed acyclic graph.
21: Call preprocess (); Call Training (); Call Prediction ()
22: //Call Confusion Matix () + Curve Analysis Receive Operating characteristics ()
23: QoS = Kubeflow Pipelines. All ()
24: QoSfactorial: = 1 … … … .1 + n
25: Set. counter: = 1 … … … … 1 + n
26: While counter < QoS.Metrics
27: QoS.factorial: QoS.factorial + counter
28: End if
29: End procedure

3.8 | CPS linear routing algorithm

This section describes the second step algorithm for determining the most suitable minimum spanning tree for VANET nodes. The LRA is a graph routing scheme for VANET entities. Algorithm 2 shows the LRA reconfiguration for where Line 1-2 establishes the VANAET entities. Line 3-15 locates and allows all the valid VANET entities to be managed while transforming data streams for analytics.

Algorithm 2.

Let $G = (V, E)$ be a connected graph with $V = \{1 \ldots \ldots .., n)\}$, let $w : E \rightarrow R$ be a linear weight function for which $n+1$ distinct VANET edges always have distinct weights.
1: Procedure ( )
2: VANET $(G, w; T)$
3: For $s = i = 1$ *to* n do $V_i \leftarrow \{i\}$ *od*;
4: $T \leftarrow \emptyset; M \leftarrow \{V_1 \ldots \ldots \ldots \ldots V_n\}$;
5: ***While*** $|T| < n-1$ ***do***
6: ***For*** $U \in M$ ***do***
7: Find an edge VANET $V_n = uv$ with $u \in U, v \notin U$ *and* $w(e) < w(\acute{e})$
8: For all edges $\acute{e} = \acute{u}\acute{v}$ with $\acute{u} \in \acute{U}, \acute{v} \notin U$;
9: Call Transform ( )// Extract data streams for structured datasets
10: Find the component VANET $\acute{U}$ remaining v;
11: $T \leftarrow T \cup \{e\}$;

12: Call Analytics ( ) // Visualization on Cloud dashboard
13: ***end***
14*:* *od*
15: ***For*** $U \in M$ ***do*** **MERGE** $(U\acute{U})$ od
16: Call cloud orchestration ( ) // manage all structured datasets
17: Call QoS.factorial ( )
14: ***od***
15: ***End***

3.9 | Complexity of algorithm

Considering Algorithms 1 and 2, the identified complexities in the system algorithm are Time and space complexities. For Kubeflow-pipelines, I/O data streams are trained and learned based on scheduling orchestration services within the edge, fog, and cloud. These have task generation, deliverable QoS, resource utilization. The key actors are the edge, fog, and cloud microservices. Basic configurations are made for all these processes. For the service operations, the space complexity depends on the order of $O(n)$ while the time complexity is associated with the order of

$$f(n) \rightarrow 2n^3 + 4n^2 + 2n,$$

$$O\ (Algorithm\ I.) \rightarrow O\left[2\left(n^3\right) + 2\left(n^2\right) + n\right]$$

$$\rightarrow O(nlog\ n),$$

hence, time complexity is $T(n) \rightarrow O(nlog\ n)$.

The space complexity has various instances of independent actors namely, edge = 1, fog = 1, Kubeflow pipeline = 1, QoS = 1

As such, the space complexity is $s\ (n) \rightarrow O(n)$.

4 | CYBER-PHYSICAL IMPLEMENTATION SCENARIO

In this section, we shall apply the previously discussed algorithms while introducing the AI engine which is responsible for full-stack integration as shown in Algorithm 1. Datastream provisioning from the edge to the cloud will be highlighted.

The experimental details of iFog cloud datacenter scalability and the redundant replication characteristic model for the CPSS Spine-leaf architecture will be discussed later in this section.

As for implementing the AI engine, Figures 10 to 16 is achieved by integrating Cloud tracking application/GPS web portal based on Google mapped services. The idea is to track location-based objects. Python 3, java server page (JSP), and MySQL database are employed. The various output results are determined. The AI algorithm in Figure 7 learns and handles security issues and identity management on the web portal login interface. As shown, the web portal login interface requires a login username and password. Upon login, a dashboard interface offers the content of the application and all the menu assigned to the User. Once the initial PMT Check box is launched, an expanded version is shown in Figure 8. It explains how to access the movement details of any range of vehicles.

To access any range of vehicles for the movement details, this is achieved by clicking on any vehicle range to expand it. Table 2 shows the vehicles within that range from Figure 11.

Figure 12 shows the location snapshot of a selected vehicle at a particular time in map view. Figure 13 shows the screenshot depicting a selected Vehicle's location at a particular time in map view. Figure 14 shows the screenshot displaying how to search and fetch a vehicle's movement or trajectory details (historical data report) using a date and time range.

Figure 15 shows the report of the entire route taken by a vehicle for a particular journey from start to finish. The responses leveraged unsupervised trajectory mining methods. Figure 14 shows the traffic distribution of time intervals and distance intervals between two consecutive points. The average sampling interval is about 177 seconds, with a distance

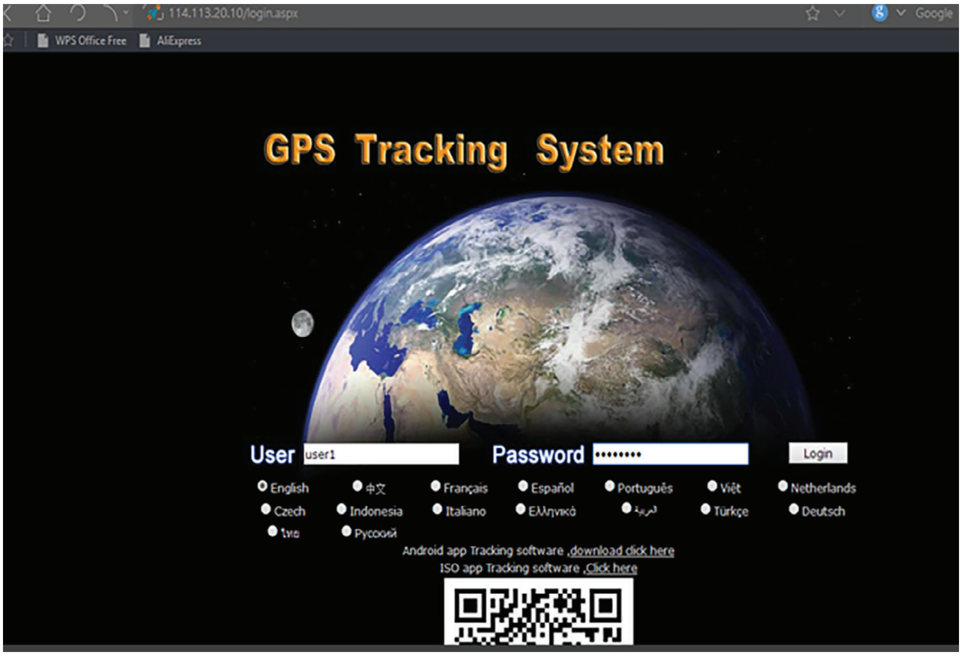

FIGURE 10 CPS VANET identity management control

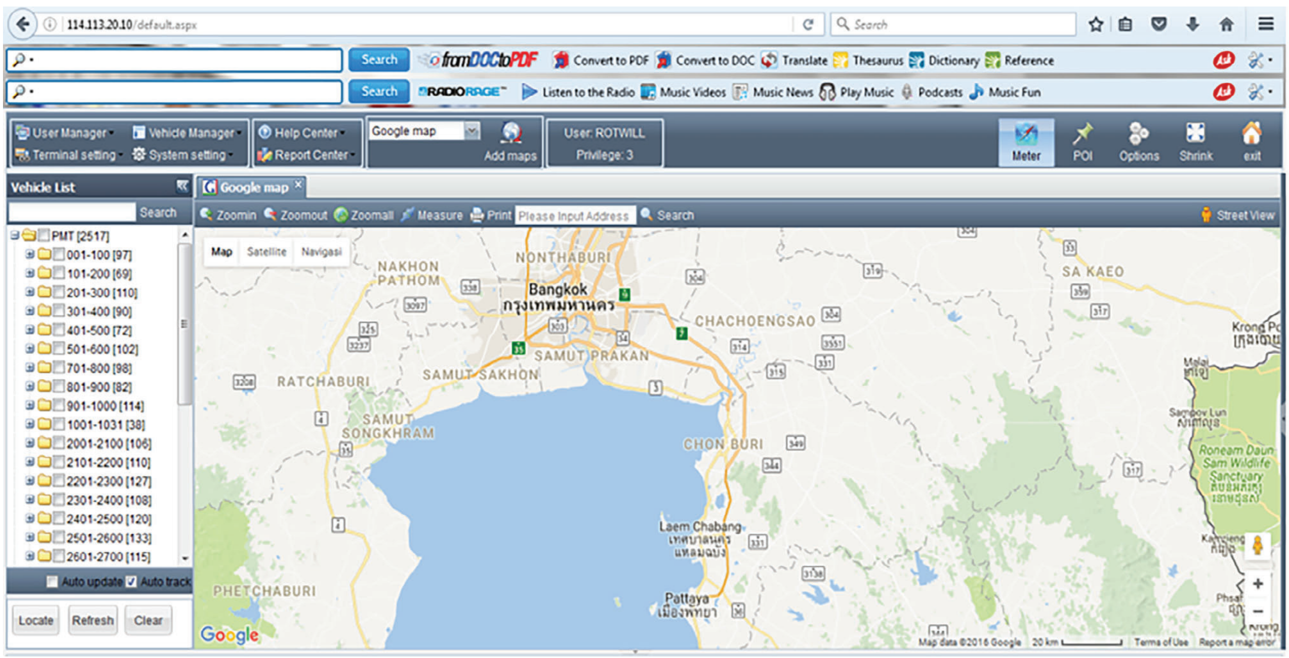

FIGURE 11 CPS VANET access V2V movement

of about 623 m. Each sample of this dataset which is named by the vehicle ID contains the predefined trajectories. With the GPS interface, the instantaneous trajectory coordinates at any spot or location near Enugu province of Enugu-State and the other traffic elements of the vehicles. By placing the mouse over the map on motion, the coordinates points are reported. This may be an estate, housing, tourist attractions, hotels, cottages, or other interest information. A note can be taken on the exact coordinates of that point. From the huge datasets obtained, a representative plot is shown in Figure 14.

So far, this work has discussed the PMT use case scenario. Considering the haulage CPS, a trajectory tracking drive test is used to obtain enormous datasets for the AI model. This dataset contains the GPS trajectories of 10 357 peace mass

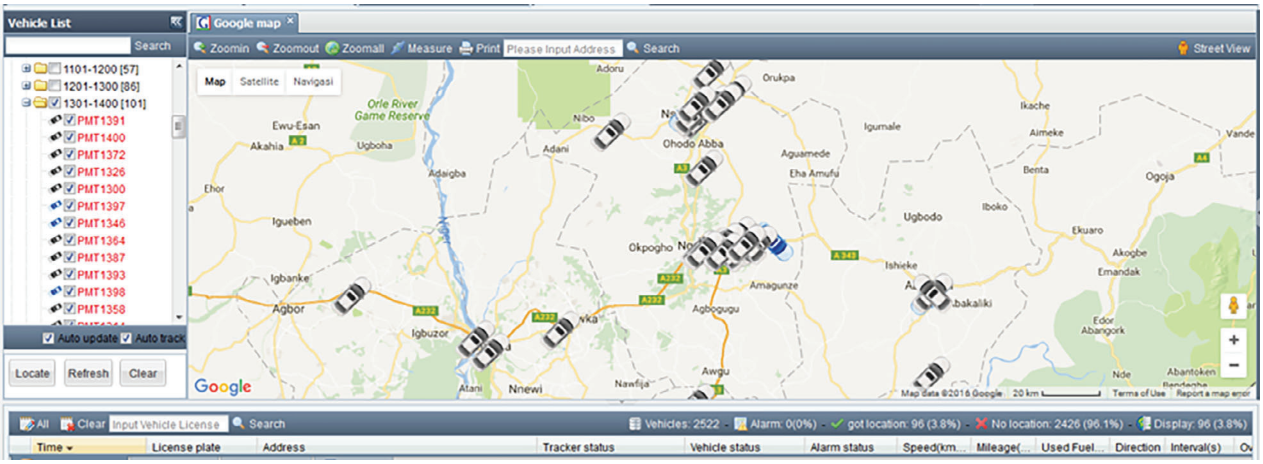

FIGURE 12 CPS VANET V2V location map view

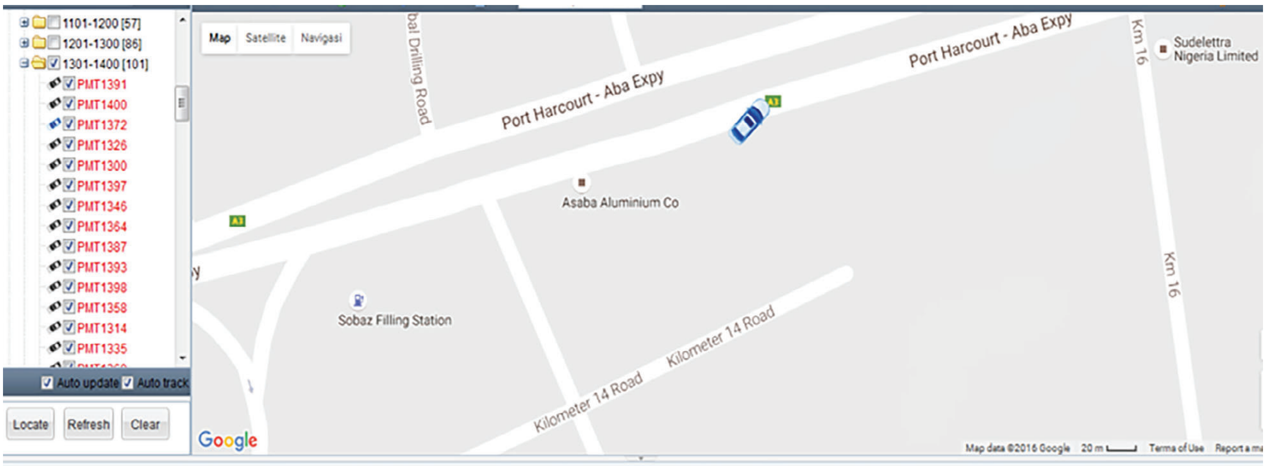

FIGURE 13 CPS VANET V2V map view location selection snapshot

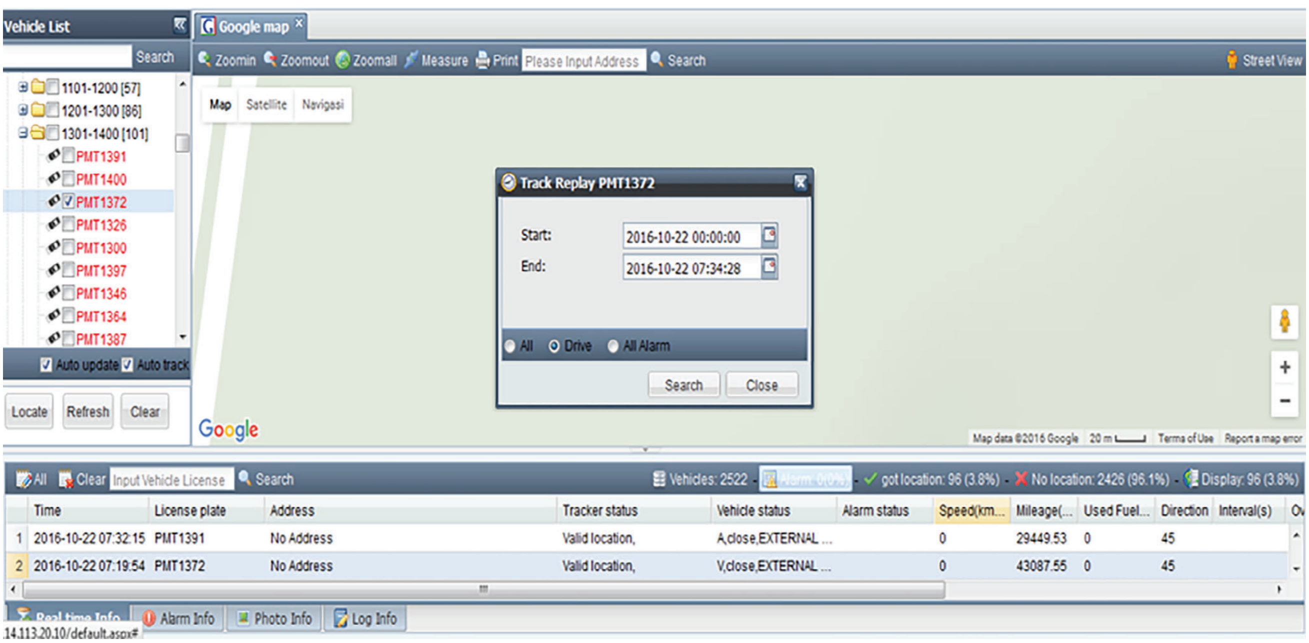

FIGURE 14 CPS VANET V2V search function

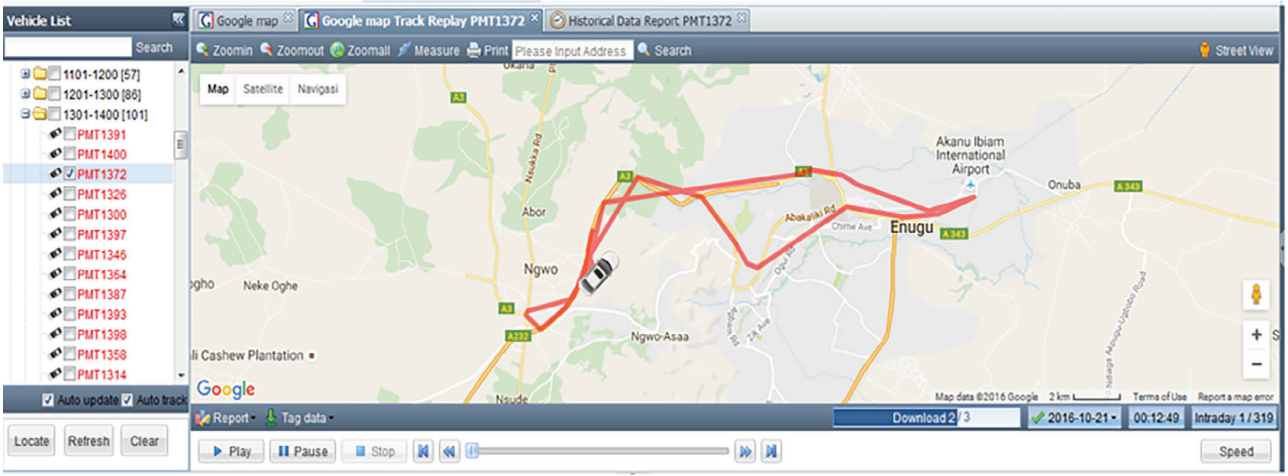

FIGURE 15 CPS VANET V2V end to end routes

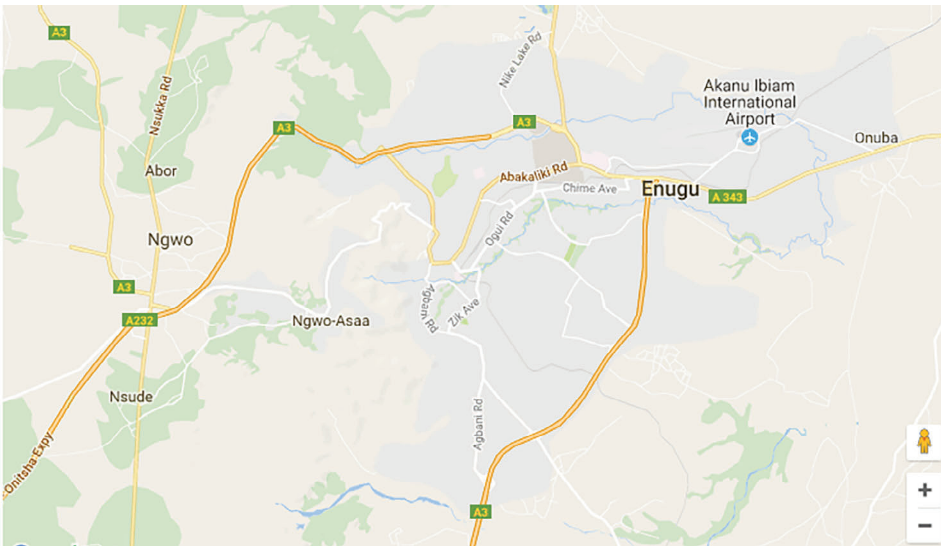

FIGURE 16 CPS VANET V2V trajectory pathways

vehicles from February 2013 to 2014 within Enugu State, Nigeria. The total number of points in this dataset is about 15 million, and the total distance of the trajectories reaches 9 km via GPS interface modules. Figures 17 to 19 shows the geospatial longitude and latitude of a few selected vehicular IDs.

5 | EXPERIMENTAL SIMULATION

5.1 | Scenario description

This section considered the use case performance of CPS architecture using numerical simulations[26] via docker engine. iFogSim experimental platform, which offers real-time scenarios for CPS Fog-driven networks for the smart city domain, is employed, as shown in Figure 20. Additionally, the tuning parameters used in a similar network design[47] are explored. iFog middleware is deployed as neural virtual machines (NVMs) used to handle workload requests. Poisson distribution with an intelligent response time policy (iRTP) is used in configuration profile management. From the cloud, the incoming traffic is referred to as command tasks, for example, web-based queries. The QoS computational algorithm lookout for

TABLE 2 CPS VANET V2V historical (trajectory) data report

Time stamp	Licence plate	Address	Tracker status	Vehicle status	Speed (km)	Milage	Usage fuel	Direction
October 22, 2016 07:17:11	PMT 1312	7 Murtala Mohammed Way, Lagos, Nigeria	Valid location	A, Start, External	0	4 157 458	0	135
October 22, 2016 07:16:08	PMT 1392	No address	Valid location	V, close, external	0	24 736.03	0	45
October 22, 2016 07:16:02	PMT 1357	No address	Valid location	A, Close, External	0	48 393.72	0	45
October 22, 2016 07:15:44	PMT 1397	No address	Valid location	V, Close, External	0	41 604.71	0	45
October 22, 2016 07:15:40	PMT 1361	No address	Valid location	V, Close, External	0	8474.47	0	45
October 22, 2016 07:16:52	PMT 1335	No address	Valid location	A Close, External	55	37 133.74	0	225
October 22, 2016 07:15:22	PMT 1391	No address	Valid location	A, Close, External	0	24 449.53	0	45
October 22, 2016 07:14:28	PMT 1374	No address	Valid location	A, Start, External	7	35 699.18	0	315
October 22, 2016 07:18:18	PMT 1385	No address	Valid location	A Close, External	0	36 825.77	0	45
October 22, 2016 07:16:30	PMT 1371	No address	Valid location	A, Close, External	0	48 755.83	0	45
October 22, 2016 07:11:56	PMT 1376	No address	Valid location	V, Close, External	0	41 275.39	0	45
October 22, 2016 07:11:52	PMT 1356	No address	Valid location	V, Close, External	0	21 491.14	0	45
October 22, 2016 07:18:36	PMT 1361	No address	Valid location	V, Start, External	0	33 629.19	0	255
October 22, 2016 07:11:00	PMT 1372	No address	Valid location	V, Close, External	0	43 087.56	0	45

resource types (pool CPU and intra-inter network resources). These are used for carrying out I/O data streams in the smart city ecosystem. Considering the overall traffic workload, the system evaluation is done with a C++ program that simulates a network of IoT, fog, and cloud servers, as depicted in Figure 1. The system reads the traffic trace files and generates arbitrary traffic patterns based on a discrete-time Markov chain (DTMC). This comes in the form of Poisson distribution. Overall, the system is implemented using spine-leaf fog network simulator with AI sandbox (SLFNS-AI Sandbox) depicted in Table 3. The idea is to see how to reduce network latency, and any possible network congestion issues in a highly distributed and multilayer virtualized IoT-Fog-Cloud environment. The implementation is done with a discrete-event C++ library to simulate fog cloud entities and services.[26] The analysis consists of a CPS data center with the following configuration but similar to the distributed cloud[47] namely, Xen as the virtual machine manager (VMM), x86 architecture, Linux operating system, and one virtual machine (VM) powered with 2048 MB RAM and a host having 16 384 MB of RAM. Figure 1 shows that the critical device deployment locations are found in the edge, fog, and cloud layers, respectively. These are the essential layers in the six-layered architecture considered for the IBM cloud platform.[48]

5.2 | Case study for iFog VANET spine-leaf architecture

In this section, we shall discuss how to integrate selected specialized TCP/IP VANET nodes for end-to-end traffic management in SLFNS-AI Sandbox. Also, the validation of the proposed iFog VANET spine-leaf architecture for optimal traffic case scenarios will be discussed.

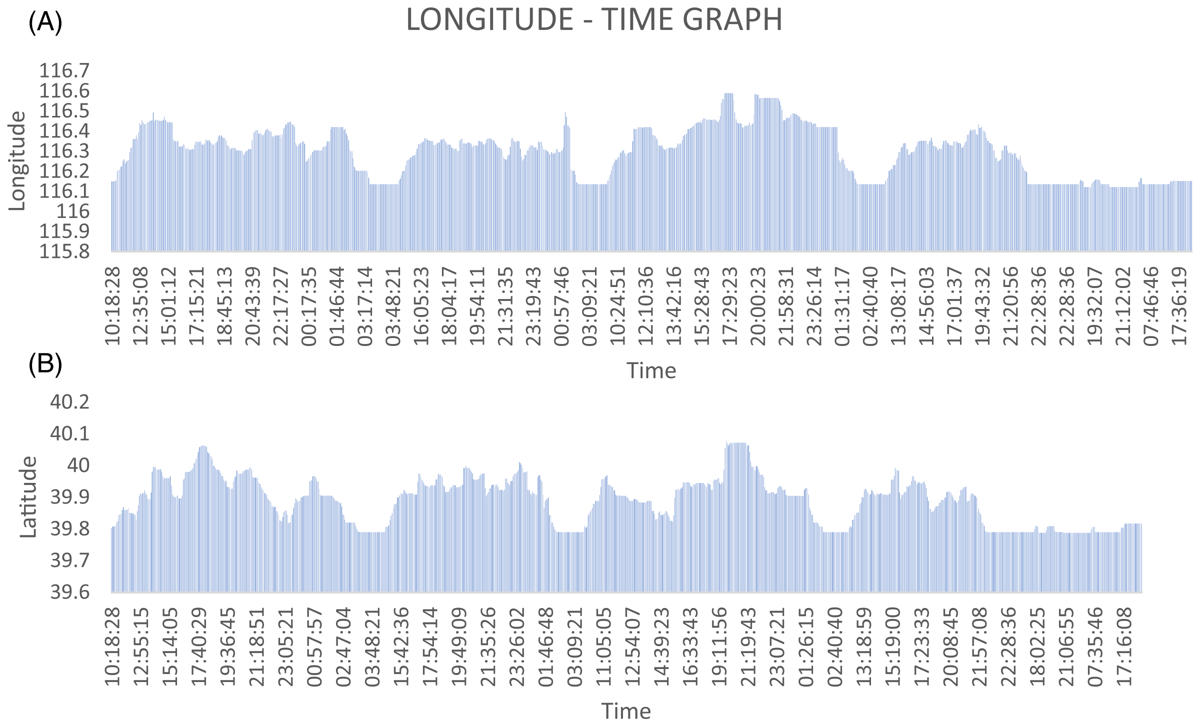

FIGURE 17 (A) Longitude time graph for vehicle ID PMT 197. (B) Latitude time graph for vehicle ID PMT 197

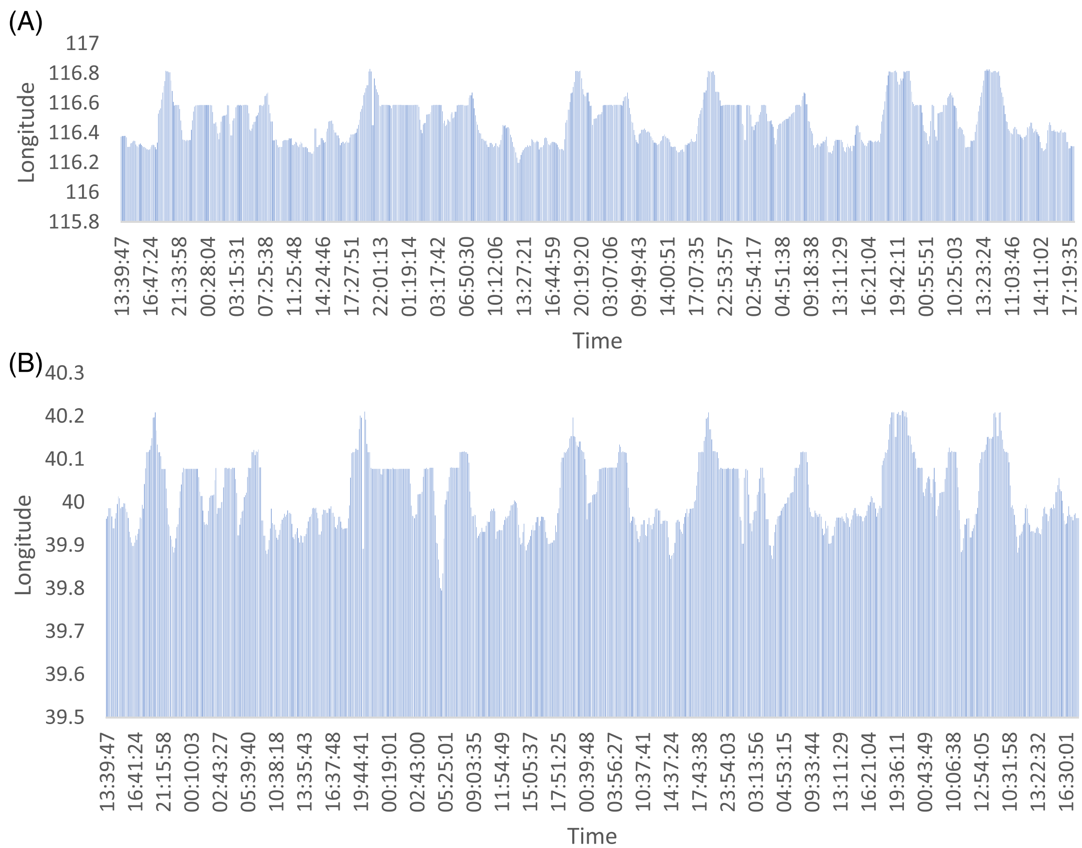

FIGURE 18 (A) Longitude time graph for vehicle ID PMT 167. (B) Latitude time graph for vehicle ID PMT 167

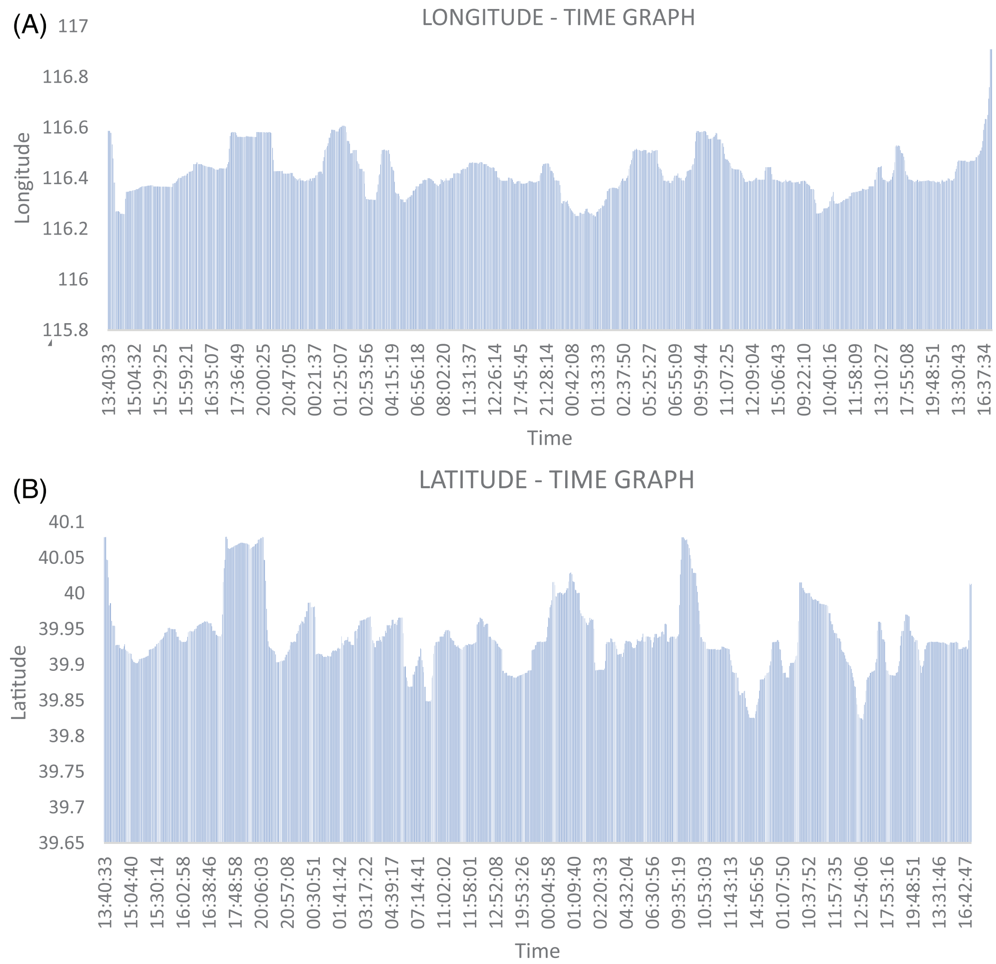

FIGURE 19 (A) Longitude time graph for vehicle Id PMT 155. (B) Latitude time graph for vehicle Id PMT 155

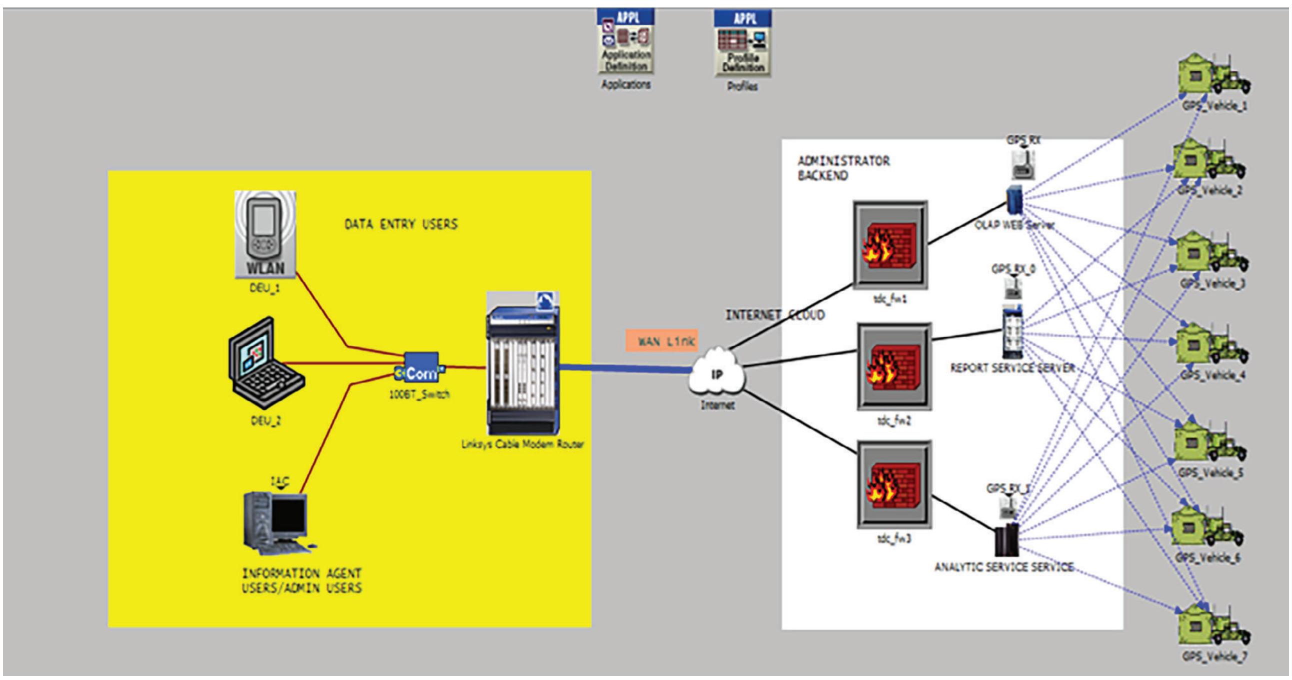

FIGURE 20 Global simulation model for haulage decision support system

TABLE 3 Simulation specifications

Simulation parameter	Value
Riverbed modeler C++	Spine-leaf fog network simulator with AI sandbox (SLFNS-AI sandbox)
Discrete event time	1000 s
Area	1000 m * 1000 m
Number of VANET modes	1000 (constant bit rate link)
Data payload	512 byes (packet size)
Transmission range	100 m
Traffic distribution	Poisson distribution with an intelligent response time policy (iRTP)
Network backbone	MPLS, IPv6 backbone 5G/LTE/Wi-Fi
iFog middleware	Neural virtual machines (NVMs)
Operating system	Docker engine

First, trajectory data for haulage movement is designed while integrating GPS for mapping location entities such as longitude, latitude, date, and time. This forms the logistic communication network depicted in Figure 20. The design comprises data entry users connected via a 100 MB Ethernet switch to a Linksys cable modem router. The system is connected to a WAN link vis-à-vis the IP cloud. The administrator's backend housing online analytical processing (OLAP) web server, report service server, and analytic service server. A GPS transceiver is connected between the servers and the vehicles for full-duplex communication of events. The network communication runs via MPLS, IPv6 backbone illustrating the end-to-end logistic behavior. The rest of the work will show the experimental details of iFog cloud datacenter scalability and the redundant replication for the CPSS spine-leaf architecture needed to support Figure 5.

Figure 21 reveals that 83.33% of the traffic workload is utilized at the fog layer, while 16.67% is consumed in the cloud layer. At the fog, resource discovery is based on the peer-to-peer transactional (events and commands) exchanges involving resource allocation between the Fog and the cloud. Workload utilization responses at both the fog and cloud domain are shown in Figure 21, but the fog computing layer is observed to house several events streams, thereby accounting for higher traffic intensity (ie, 83.33%). Hence, for smart city applications, heterogeneous services will utilize more resources at the Fog than similar consumption on the cloud. It is further noted that at the fog layer, the entire data stream processing takes place via iFog gateways. Hence, the data requirements pushed to the cloud gets reduced greatly. This might become incredibly challenging when dealing with massive stream analytics.

Therefore, an intelligent (AI) algorithm must be orchestrated to avoid system collapses during peak workload provisioning.

Figure 22 shows the latency behavior observed considering the traffic workload at both the Fog and cloud within the iFog spine-leaf architectures. Within the dynamic Fog computing architecture, traffic originating from the edge explores cloud computing Infrastructure's benefits as a Service (IaaS). With the telemetry data stream generated, this results in high

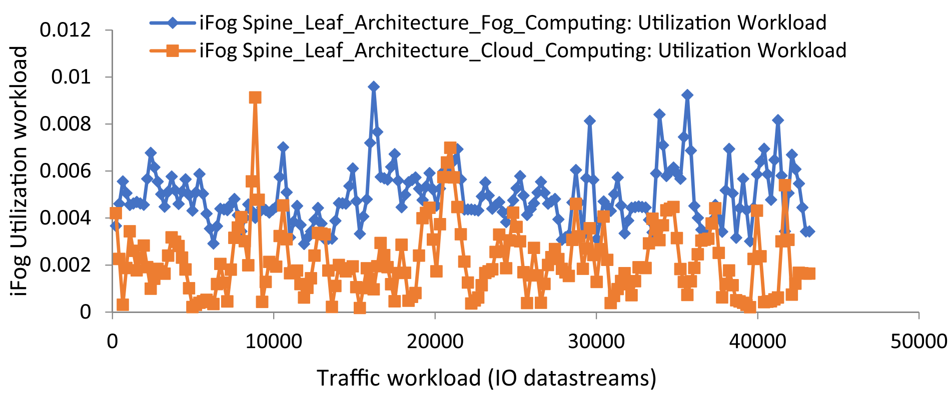

FIGURE 21 iFog utilization workload for traffic workload at the time ($t > 0$)

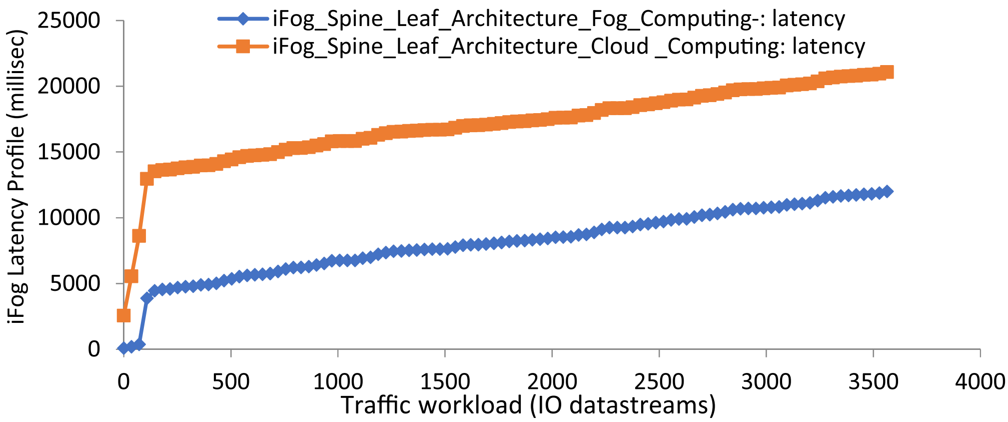

FIGURE 22 iFog latency profile for traffic workload at a time ($t > 0$)

data traffic capable of inducing streams congestion. It is observed that round-trip time delay resulted from volumetric data movements and hop counts from the edge to the cloud. At the fog, time-sensitive services had lower latency, satisfying the minimum latency concerns of most complex networks. Data streams communication, computation, and network latencies at the Fog stimulate smart processing, storage, and stream analytics at the Fog. The results indicated fog and cloud streams latency at 20.31% and 77.69%, respectively. This is realistic for complex network ecosystems such as VANET.

5.3 | Case study for iFog VANET spine-leaf congestion control

This section presents the analysis of the main parameters of the V2V simulation design for congestion monitoring in Figure 23 (proposed VANET Architecture). This work established a comparison with VANET linear routing algorithm alongside other traditional collection tree protocols (CTP),[46] LEACH algorithm.[49] The algorithmic performance considerations are checked under three parametric service provisioning metrics across the VANET network model: resource

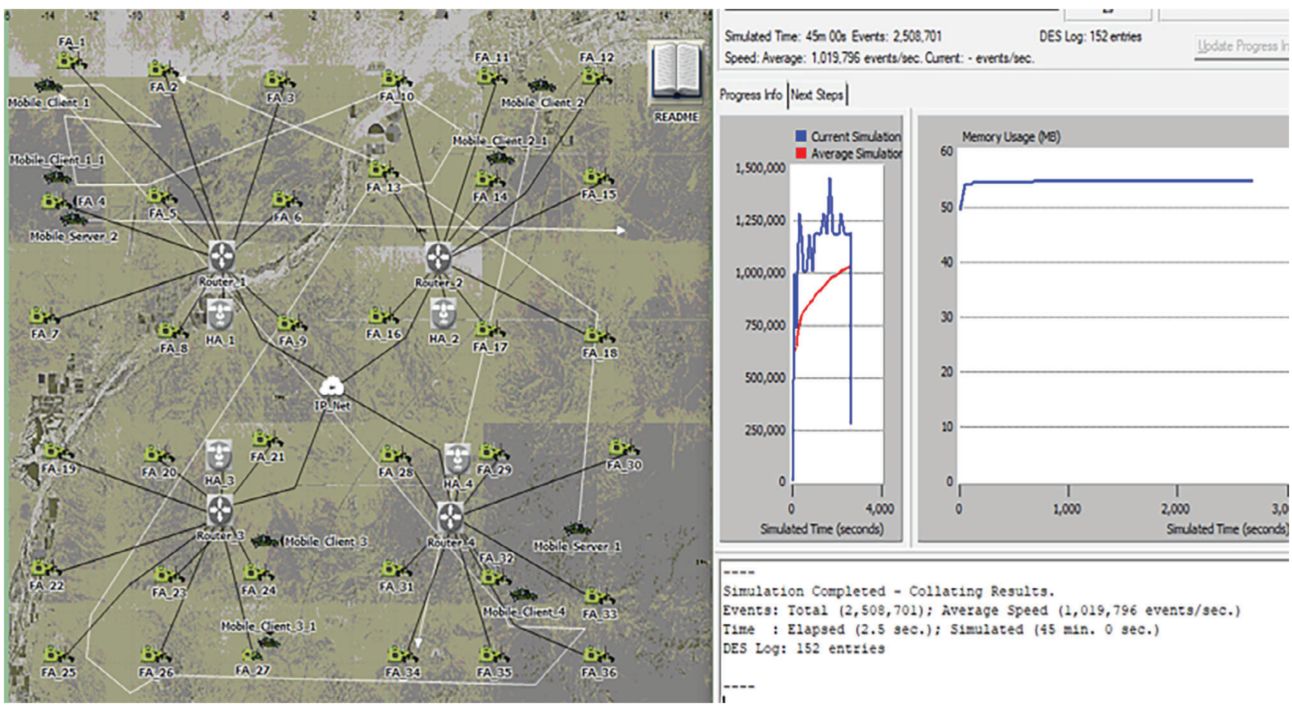

FIGURE 23 V2V simulation characterization and successful execution

utilization, latency, and throughput. In the simulation design, the V2V heuristic decision-making is constructed, as shown in Figure 23.

To assess the performance of the proposed schemes, three TCP/IP congestion control schemes, including CTP, LEACH, and LRA, are simulated as shown in the experimental setup. For all the V2V scenarios, an analysis of the performance of these three mechanisms in terms of network resource utilization, latency, and throughput response is discussed.

Figure 24 shows the simulated response of the network utilization. The average transmission rate of source nodes under different schemes is shown. The plots indicate the intensity of resource utilization for the VANET clusters. LRA, LEACH, and CTP had resource utilization of 34.45%, 32.18%, and 33.37%, respectively. The proposed LRA provides higher resource utilization in terms of CPU, memory, I/O, among others. This, however, happens with very low latency for service provisioning.

Figure 25 shows the latency response of the V2V model. The performance of the three schemes indicates that the CTP offered the lowest latency and showed less fluctuation in the steady-state than the others, hence offering the shortest link for vehicular movement and tracking. LRA, LEACH, and CTP had a latency of 11.76%, 82.35%, and 5.89%, respectively.

In Figure 26, the network throughput is the average rate of successful message delivery over a Vehicular communication channel measured in bits per second (bit/s or bps) or bytes/seconds. It is the sum of the data rates that are delivered to all nodes in a network. While the throughput scenario for LRA, LEACH, and CTP offered a similar response 19.61%,

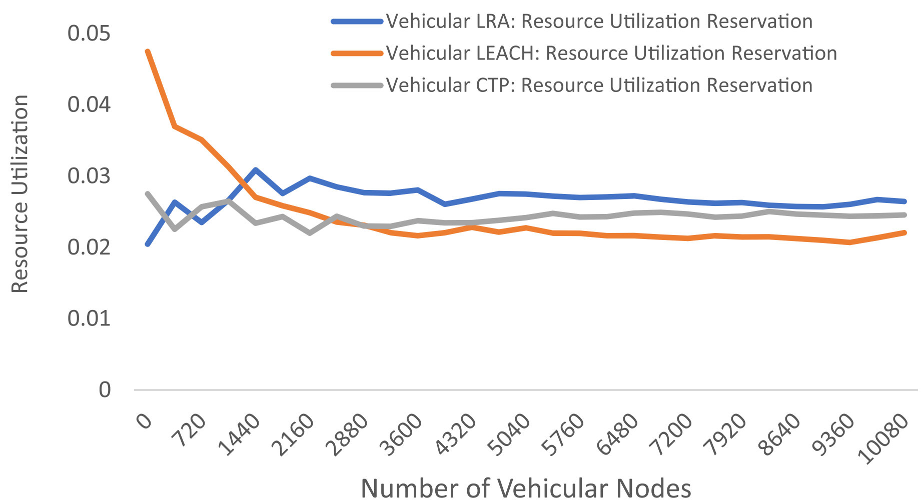

FIGURE 24 V2V resource utilization

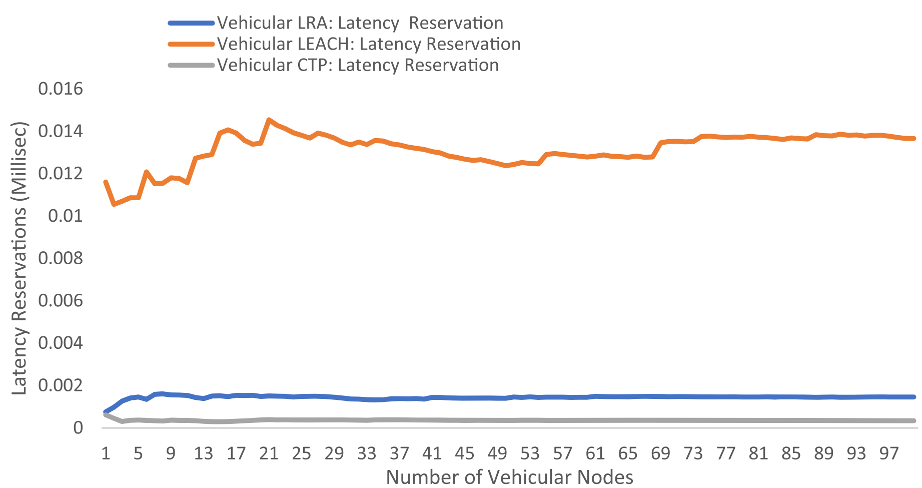

FIGURE 25 V2V latency reservation response

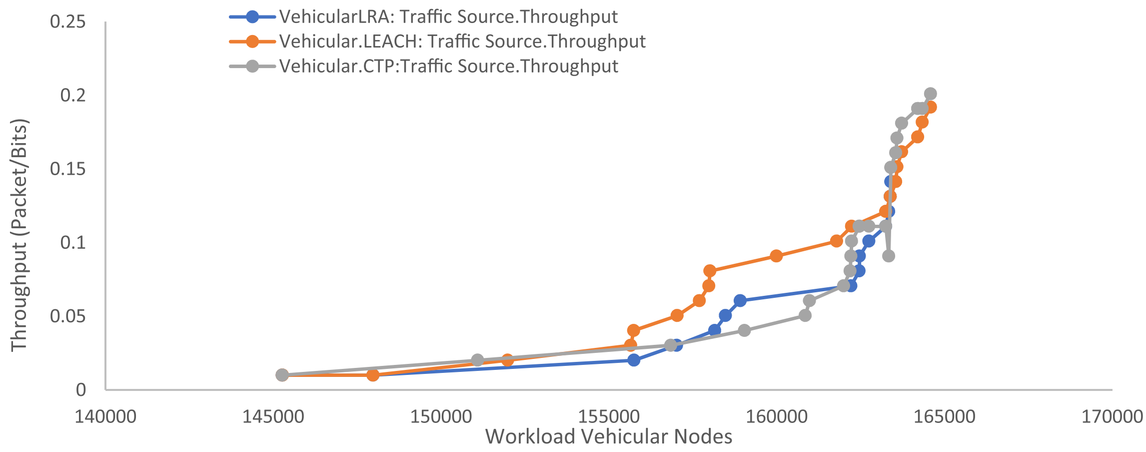

FIGURE 26 V2V throughput response

39.22%, and 41.17%, respectively. The results show that the throughput with LRA is relatively satisfactory compared to the other schemes, especially when the bit error rate becomes smaller. This accounts for a better deployment in a high-density sensing environment.

6 | ANALYSIS OF KUBEFLOW PIPELINE MACHINE LEARNING

Recall from Section 3.4; we discussed trajectory data clustering and mining (TDC-DM) servers between various VANET clusters. Also, a supervised learning algorithm is used in the training data of the vehicles. This produced an inferred function and is used for mapping new trajectory patterns in the respective servers. An optimal scenario allows the algorithm to determine the class labels for unseen instances and patterns correctly. The neural cluster data mining technique for knowledge discovery in the trajectory database is fully implemented using SOM Kubeflow pipelines in the software design. This ensures that the design visualization that maps the expected trajectory given to the haulage driver (for the actual trajectory taken by the haulage driver) facilitates decision-making. As depicted in Figure 20, the process of model design with alternative routes in the large-scale transportation transit system includes two main components:

1. The heuristic decision-making construction according to the general scheme described.
2. Intelligent method for transformation of heuristic decision-making.

Figure 23 illustrates the trajectory dataset selection for training from the clustering neural network. The datasets are loaded into matrix columns from the workspace and compiled for the clustering training. The Trajectory dataset selection for training is depicted in Figure 27. This shows the neural network architecture that maps the input datasets to the clustered pattern via the SOM layer. Now, the SOM is defined into a dimensional map of 10 for simplicity

Figure 28 depicts the trajectory dataset (training samples). The Algorithm I trains the network to learn the topology and distribute the input samples using SOM. Training is dynamically enabled to stop when the full number of epochs has occurred. The figure shows the trajectory iterations for the dataset under training. The Kubeflow pipeline platform is leveraged to achieve the SOM while providing supports for management and tracking VANET jobs and runs. This provides the engine for scheduling multistep ML workflows while offering an SDK for defining and manipulating pipelines and components. The main benefits in this work include:

- End-to-end VANENT orchestration: enabling and simplifying the orchestration of machine learning pipelines.
- Easy experimentation of VANET workload trials

Easy reuse allows for component reuse and pipelines to quickly create end-to-end solutions without having to rebuild each time.

Figure 29A demonstrates the results of SOM when there is no data mining in the datasets. This shows a similar hierarchical model with the existing peace mass model discussed in the previous sections of unsupervised trajectory mining methods. Figure 29B shows the SOM with data mining for all the neighbor connections (vector inputs) in peace

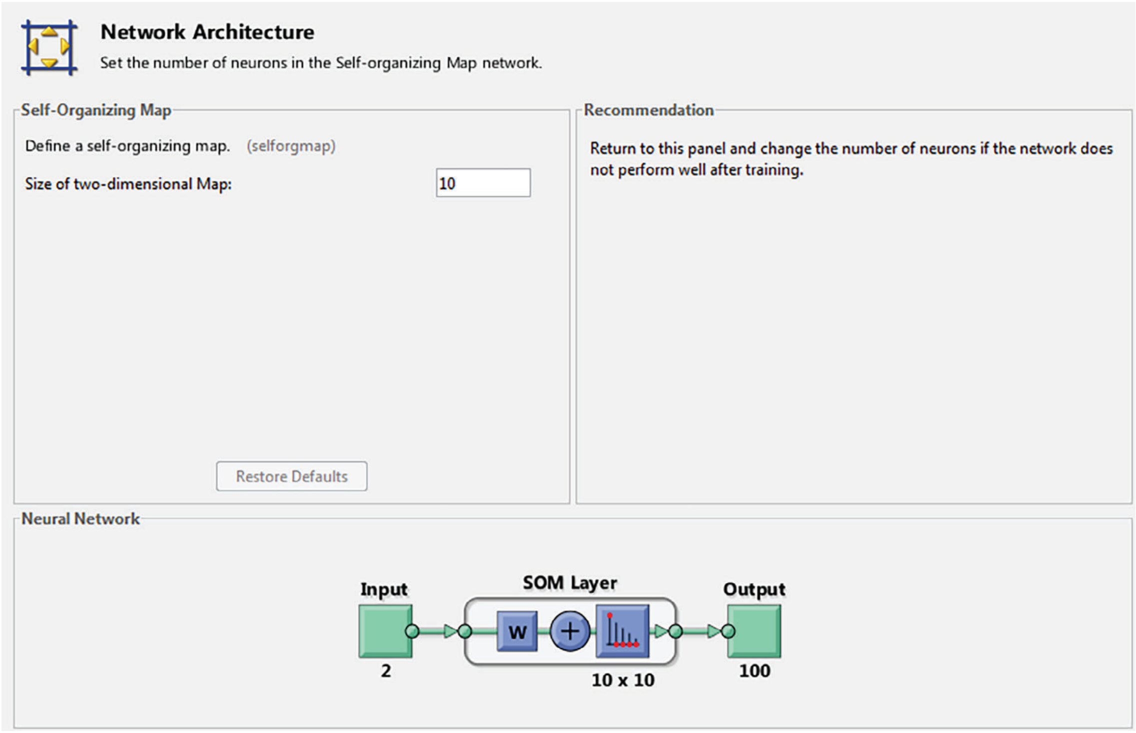

FIGURE 27 Trajectory dataset network architecture by SOM

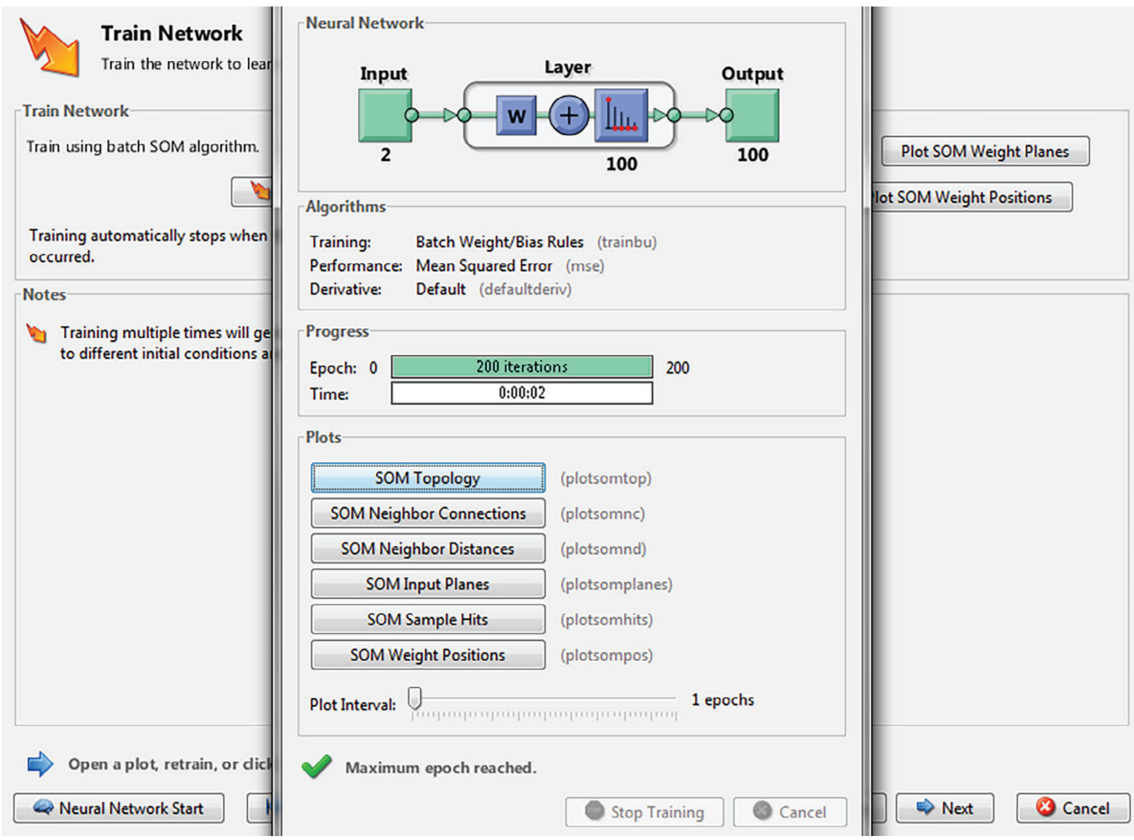

FIGURE 28 Trajectory dataset training iterations

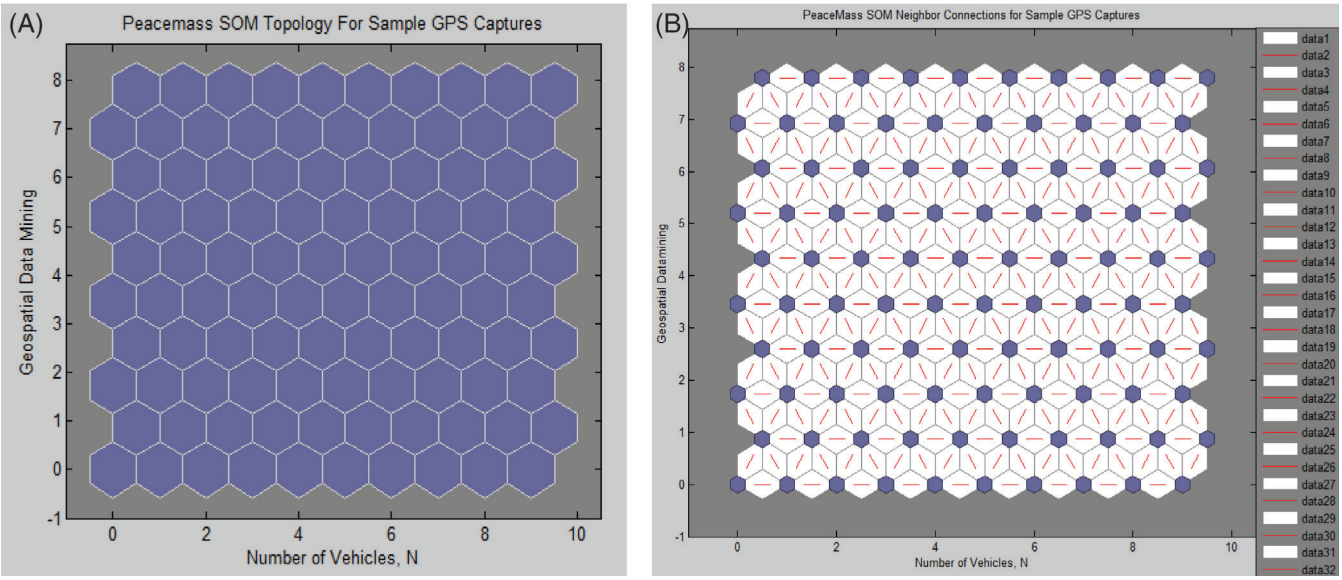

FIGURE 29 (A) SOM with no-data mining in peace datasets. (B) SOM with data mining with neighbor connections in peace datasets

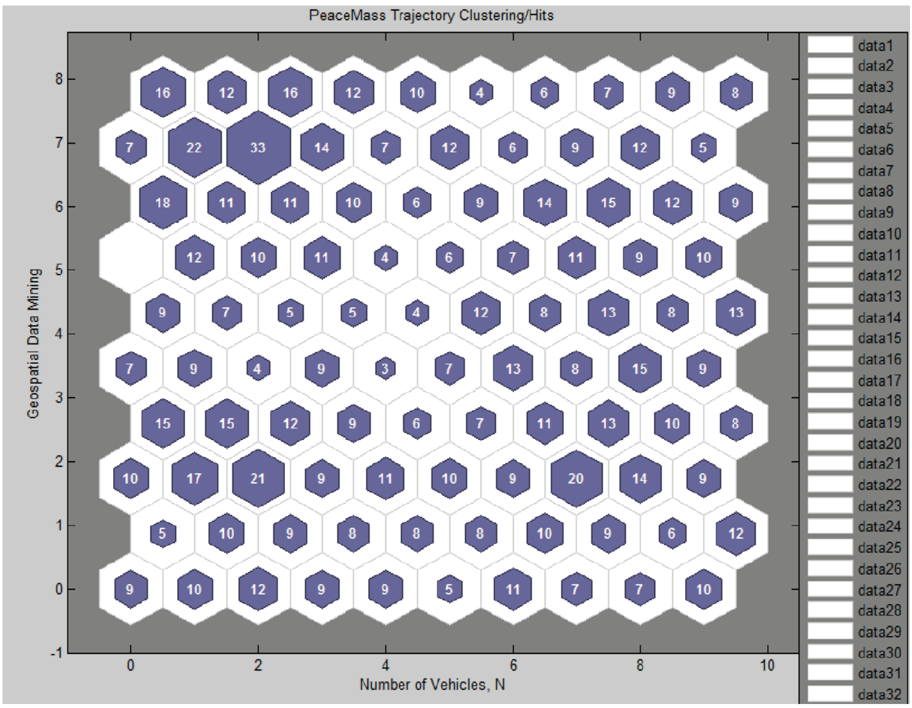

FIGURE 30 SOM with data mining with neighbor clustering in peace datasets

datasets, while Figure 30 shows the SOM with data mining for neighbor clustering in peace datasets. These satisfy the requirements of supervised trajectory mining.

7 | IBM USE CASE SCENARIO DEPLOYMENT FRAMEWORK

So far, this work has discussed the traffic workload implications based on QoS metrics such as streams latency and resource utilization. In this section, a use case deployment description with IBM Cloud is presented. All AI-driven applications must be tested on a Cloud network platform to ascertain resilience and workload scalability, as shown in Figures 31 and 32, respectively. In this regard, this work considered the cloud-based IBM infrastructure as a complete stack.

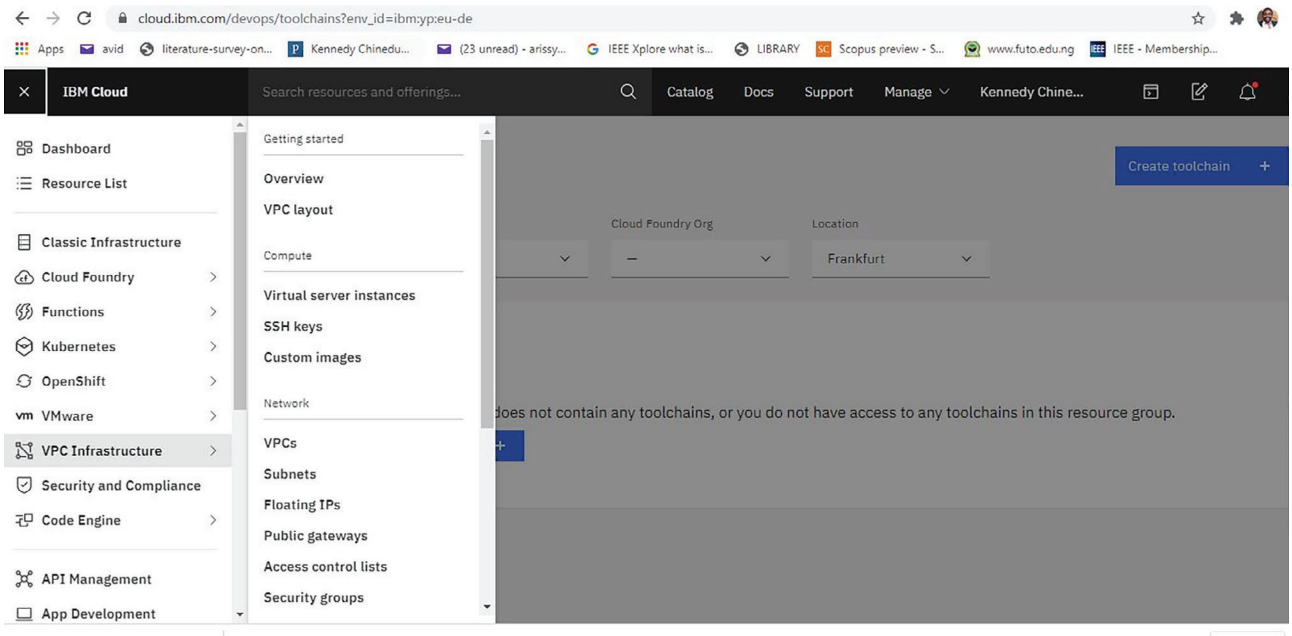

FIGURE 31 IBM cloud workload for traffic workload at the time (t > 0)

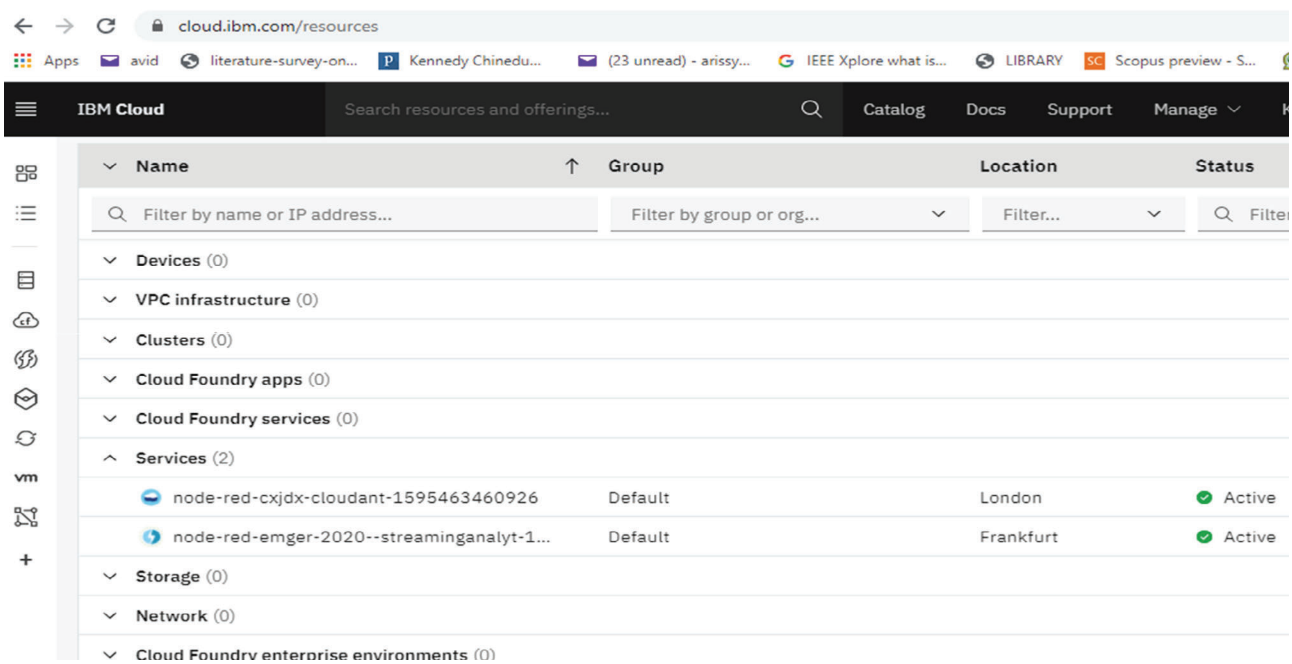

FIGURE 32 IBM cloud fog workload for traffic workload for nodes ($n = 2$)

This is a public deployment offering with immersive services located in the catalog platform. Compute storage, platform networking, developer end-to-end application development leveraging IBM's numerous services, and open platform makes it ideal for all cloud-native integrations. For the CNCDS on which Society 5.0 runs, cloud Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) are fused for quick testing and agile supports. The multicloud delivery model leverages open-source Kubernetes, and Red has OpenShift, virtual machines, containers, bare-metals, and server-less to introduce control/flexibility into the data stream workloads. The introduced Society 5.0 in this research is currently deployed as Cloud-native apps with workload portability. The benefits of leveraging IBM Cloud include:

i resilient console for full-stack implementation
ii Identity and access management model for access control

iii the product supports IBM Catalog.
iv Tagging and searching schemes for isolating and determining resources.
v accounting and billing systems with security modules.

8 | CONCLUSION AND FUTURE WORK

This research presents an AI-driven critical infrastructure (iFog) for smart cyber-physical social systems from theoretical and real-world perspectives. The architecture is optimized for AI integration, especially for edge-to-cloud transactions. It highlights how both structured and unstructured VANETs datasets could be aggregated for analytics. A container-based IBM cloud foundry Kubernetes (production environment) is used to depict the real-world analytics engine. The iFog spine-leaf implementation is presented to offer scalability and optimal resource usage, especially for smart city workloads. We presented iFog optimization model for scalability and QoS dynamic provisioning. The work highlighted feasible integration for datastream workload. Resource utilization for the computational processes shows that the Fog layer carries out most tasks at about 83.33%, while the cloud handles about 16.67% of the workload.

Similarly, results show fog and cloud stream latencies at 20.31% and 77.69%, respectively, for complex network ecosystems like VANET. The results reveal that the iFog computational model enables faster processing time and reduces network overhead for real-time applications due to high resource provisioning. Also, LRA showed promising QoS potentials as it offered optimal utilization, latency, and throughput responses for real-time deployment. Consequently, data streams communication and computation are better provisioned at the Fog with the least overhead. Future work will focus on real-time implementation with IBM Watson and cloud foundry platform for IoT-edge devices integrations. The areas of big data analytics will be investigated using relevant AI algorithms.

8.1 | Future challenges and research opportunities

Some identified CPS challenges, especially with the advent of AI, include:

i **Complexity of cyber-attack monitoring**: The development of a comprehensive AI security in CPS can be very challenging as this involves the creation of dynamic, intelligent agents that can automatically determine any form of attack vectors.
ii **CPS infrastructural design**: CPS involves complex high-performance computing cloud infrastructure. This requires technical and supportive operational technologies. It must support AI algorithms loaded with computational resources. This calls for HPC design that is efficient, scalable, and fault-tolerant.
iii **Complex analytics**: The CPS for AI data stream must address complex problems efficiently. Clearly, as the number of computing devices increases, the density of traffic workload increases too, making it expedient to quickly carry out data analytics. In this regard, the data analysis must be done with AI techniques on high-end computing platforms. Apache Spark and Hadoop are the legacy cyber traffic engines.[50] AI algorithms on HPCs running quantum computing remains significant in solving complex analytic problems though very complex. In summary, CPS for data stream processing requires speed and good resource management. This is where AI algorithms extract the standards and protocol needed to balance traffic workload and combat Cybersecurity threats in CPS.

DATA AVAILABILITY STATEMENT

Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

REFERENCES

1. Kumari A, Tanwar S, Tyagi S, Kumar N, Obaidat MS, Rodrigues JJPC. Fog computing for smart grid systems in the 5G environment: challenges and solutions. *IEEE Wirel Commun*. 2019;26(3):47-53. doi:10.1109/MWC.2019.1800356
2. Qin Z, Xu X, Lu B, Qian C, Wang L, Lin C. Traffic data reconstruction based on compressive sensing with neighbor regularization. *Trans Emerg Telecommun Tech*. 2020;33(2):1-17.
3. Aljumah A, Kaur A, Bhatia M, Ahamed Ahanger T. Internet of things-fog computing-based framework for smart disaster management. *Trans Emerg Telecommun Technol*. 2020;32:1-25. doi:10.1002/ett.4078

4. Qin Z, Fei X, Xie Y, Zhang Z, Li G. An improved top-K algorithm for edge servers deployment in smart city. *Trans Emerg Telecommun Technol*. 2021;32;(18):1–20.
5. Xiaohui G, Zhang G, Cao Y. Cooperative mobile edge computing-cloud computing in internet of vehicle. *Arch Energy-Efficient Workload Allocat*. 2020;32;(8):1–21.
6. Bu S, Yu FR. A game-theoretical scheme in the smart grid with demand-side management: towards a smart cyber-physical power infrastructure. *IEEE Trans Emerg Top Comput*. 2013;1(1):22-32. doi:10.1109/TETC.2013.2273457
7. Shiroishi Y, Uchiyama K, Suzuki N. Society 5.0: for human security and well-being. *Computer*. 2018;51(7):91-95. doi:10.1109/MC.2018.3011041
8. Mamduhi MH, Maity D, Baras JS, Johansson KH. A cross-layer optimal co-Design of Control and Networking in time-sensitive cyber-physical systems. *IEEE Control Syst Lett*. 2021;5(3):917-922. doi:10.1109/LCSYS.2020.3006008
9. Mahmood A, Butler B, Jennings B. Towards efficient network resource management in SDN-based heterogeneous vehicular networks. Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC); 2018:813-814; Tokyo. doi: 10.1109/COMPSAC.2018.00133
10. IEEE draft standard for framework of blockchain-based Internet of Things (IoT) data management. P2144.1/D2; July 2020:1-18.
11. Okafor KC, Achumba IE, Chukwudebe GA, Ononiwu GC. Leveraging fog computing for scalable IoT datacenter using spine-leaf network topology. *J Electr Comput Eng*. 2017;2017(2363240):1-11.
12. Okafor KC. Dynamic reliability modelling of cyber-physical edge computing network. *Int J Comput Appl*. 2019;43:1-10. doi:10.1080/1206212X.2019.1600830
13. Donassolo B, Fajjari I, Legrand A, Mertikopoulos P. Demo: fog based framework for IoT service orchestration. Proceedings of the 2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC); 2019:1-2; Las Vegas, NV. doi: 10.1109/CCNC.2019.8651852
14. Sholla S, Naaz R, Chishti MA. Semantic smart city: context aware application architecture. Proceedings of the 2018 2nd International Conference on Electronics, Communication and Aerospace Technology (ICECA); 2018:721-724; Coimbatore. doi: 10.1109/ICECA.2018.8474777
15. Prasetyo YA, Arman AA. Group management system design for supporting society 5.0 in smart society platform. Proceedings of the 2017 International Conference on Information Technology Systems and Innovation (ICITSI); 2017:398-404; Bandung. doi: 10.1109/ICITSI.2017.8267977
16. Giyenko A, Cho YI. Intelligent UAV in smart cities using IoT. Proceedings of the 2016 16th International Conference on Control, Automation and Systems (ICCAS); 2016:207-210; Gyeongju, doi: 10.1109/ICCAS.2016.7832322
17. Sholla S, Naaz R, Chishti MA. Ethics aware object oriented smart city architecture. *China Commun*. 2017;14(5):160-173. doi:10.1109/CC.2017.7942323
18. Alexandru AM, Fiasché M, Pinna C, Taisch M, Fasanotti L, Grasseni P. Building a smart maintenance architecture using smart devices: a web 2.0 based approach. Proceedings of the 2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI); 2016:1-6; Bologna. doi: 10.1109/RTSI.2016.7740632
19. López VF, Moreno MN, Chamoso P, Corchado ES, Pérez B. Mobile application for smart city management. In: Rodríguez S, ed. *Distributed Computing and Artificial Intelligence, Special Sessions. Proceedings of the 15th International Conference, DCAI 2018. Advances in Intelligent Systems & Computing*. Vol 801. Cham: Springer; 2019. doi:10.1007/978-3-319-99608-0&uscore;21
20. Brovko A, Dolinina O, Pechenkin V. Method of the management of garbage collection in the εsmart clean cityε project. In: Gaj P, Kwiecień A, Sawicki M, eds. *Computer Networks. CN 2017. Communications in Computer and Information Science*. Vol 718. Cham: Springer; 2017. doi:10.1007/978-3-319-59767-6&uscore;34
21. Liu L, Liu Y, Wang L, Zomaya A, Hu S. Economical and balanced energy usage in the smart home infrastructure: a tutorial and new results. *IEEE Trans Emerg Top Comput*. 2015;3(4):556-570. doi:10.1109/TETC.2015.2484839
22. Zhou J, Qingyang Hu R, Qian Y. Scalable distributed communication architectures to support advanced metering infrastructure in smart grid. *IEEE Trans Parallel Distrib Syst*. 2012;23(9):1632-1642. doi:10.1109/TPDS.2012.53
23. Yang C, Vyatkin V, Pang C. Service-oriented extension of IEC 61850 for model-driven smart grid automation design. Proceedings of the 43rd Annual Conference of the IEEE Industrial Electronics Society IECON 2017; 2017:5489-5496; Beijing. doi: 10.1109/IECON.2017.8216950
24. Salgues B. Innovation in society 5.0. In: Salgues B, ed. *Society 5.0: Industry of the Future, Technologies, Methods and Tools*. 1st ed. Hoboken, NJ: Wiley; 2018:157-169.
25. Shiroishi Y, Uchiyama K, Suzuki N. Better actions for society 5.0: using AI for evidence-based policy making that keeps humans in the loop. *Computer*. 2019;52(11):73-78. doi:10.1109/MC.2019.2934592
26. Naranjo PGV, Pooranian Z, Shojafar M, Conti M, Buyya R. FOCAN:a fog-supported smart city network architecture for management of applications in the internet of everything environments. *J Parallel Distrib Comput*. 2018;132:274-283. doi:10.1016/j.jpdc.2018.07.003
27. Babar M, Khan F, Iqbal W, et al. A secured data management scheme for smart societies in industrial Internet of Things environment. *IEEE Access*. 2018;6:43088-43099. doi:10.1109/ACCESS.2018.2861421
28. Xia F, Vinel A, Gao R, et al. Evaluating IEEE 802.15.4 for cyber-physical systems. *J Wirel Com Netw*. 2011;2011:596397. doi:10.1155/2011/596397
29. Kim H, Ben-Othman J. Toward integrated virtual emotion system with AI applicability for secure CPS-enabled smart cities: AI-based research challenges and security issues. *IEEE Netw*. 2020;34(3):30-36. doi:10.1109/MNET.011.1900299

30. Ryashentseva D, Vogel-Heuser B, Lueder A. Development and evaluation of a unified agents- and supervisory control theory based manufacturing control system. Proceedings of the 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE); 2018:187-192; Munich. doi: 10.1109/COASE.2018.8560539
31. Matsuda W, Fujimoto M, Aoyama T, Mitsunaga T. Cyber security risk assessment on industry 4.0 using ICS testbed with AI and cloud. Proceedings of the 2019 IEEE Conference on Application, Information and Network Security (AINS); 2019:54-59; Pulau Pinang, Malaysia. doi: 10.1109/AINS47559.2019.8968698
32. Wang F. AI and intelligent vehicles future challenge (IVFC) in China: from cognitive intelligence to parallel intelligence. Proceedings of the 2017 ITU Kaleidoscope: Challenges for a Data-Driven Society (ITU K); 2017:1-2; Nanjing. . doi: 10.23919/ITU-WT.2017.8246841.
33. Ahmed HO FLS-based collision avoidance cyber physical system for warehouse robots using FPGA. Proceedings of the 2019 6th International Conference on Dependable Systems and their Applications (DSA); 2020:262-268; Harbin, China. doi: 10.1109/DSA.2019.00040
34. Marrella A, Mecella M, Halapuu P, Sardina S. Automated process adaptation in cyber-physical domains with the SmartPM system (short paper). Proceedings of the 2015 IEEE 8th International Conference on Service-Oriented Computing and Applications (SOCA); 2015:59-64; Rome. doi: 10.1109/SOCA.2015.18
35. Liu T, Tian B, Ai Y, Wang F. Parallel reinforcement learning-based energy efficiency improvement for a cyber-physical system. *IEEE/CAA J Automat Sin*. 2020;7(2):617-626. doi:10.1109/JAS.2020.1003072
36. Wang B, Li X, Freiheit T, Epureanu BI. Learning and intelligence in human-cyber-physical systems: framework and perspective. Proceedings of the 2020 2nd international conference on Transdisciplinary AI (TransAI); 2020:142-145; Irvine, CA.
37. Vogel-Heuser B. Plenary lecture 1: cyber physical production systems/industry 4.0-challenges in research and industrial application. Proceedings of the 41st Annual Conference of the IEEE Industrial Electronics Society IECON 2015; 2015:11-13; Yokohama. doi: 10.1109/IECON.2015.7392973
38. Xu SS, Huang H, Kung Y, Chu Y. A networked multirobot CPS with artificial immune fuzzy optimization for distributed formation control of embedded Mobile robots. *IEEE Trans Ind Inform*. 2020;16(1):414-422. doi:10.1109/TII.2019.2936045
39. Liu Z, Jin C, Jin W, et al. Industrial AI enabled prognostics for high-speed railway systems. Proceedings of the 2018 IEEE International Conference on Prognostics and Health Management (ICPHM); 2018:1-8; Seattle, WA, doi: 10.1109/ICPHM.2018.8448431
40. Li H, Dong M, Ota K. Control plane optimization in software-defined vehicular ad hoc networks. *IEEE Trans Veh Technol*. 2016;65(10):7895-7904. doi:10.1109/TVT.2016.2563164
41. Zeadally S, Adi E, Baig Z, Khan IA. Harnessing artificial intelligence capabilities to improve cybersecurity. *IEEE Access*. 2020;8:23817-23837. doi:10.1109/ACCESS.2020.2968045
42. Chen J, Cheng R, Agbodike O, Lyu Y. A dynamic Backoff window scheme for machine-type communications in cyber-physical systems. *IEEE Access*. 2020;8:31045-31056. doi:10.1109/ACCESS.2020.2973239
43. Wang F, Zhang JJ. Transportation 5.0 in CPSS: towards ACP-based society-centered intelligent transportation. Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC); 2017:762-767; Yokohama. doi: 10.1109/ITSC.2017.8317905
44. Salgues B. Society 5.0, its logic and its construction. In: Salgues B, ed. *Society 5.0: Industry of the Future, Technologies Methods and Tools*. 1st ed. Hoboken, NJ: Wiley; 2018:1-21.
45. Lestari RD, Rusdinar A, Murti MA, Tawaqal G, Lee D. Design of IoT-based river water monitoring robot data transmission model using low power wide area network (LPWAN) communication technology. Proceedings of the 2019 IEEE International Conference on Internet of Things and Intelligence System (IoTaIS); 2019:201-205; BALI, Indonesia. doi: 10.1109/IoTaIS47347.2019.8980377
46. Si W, Hashemi M, Xin L, Starobinski D, Trachtenberg A. TeaCP: a toolkit for evaluation and analysis of collection protocols in wireless sensor networks. *IEEE Trans Netw Serv Manag*. 2015;12(2):293-307. doi:10.1109/TNSM.2015.2407882
47. Naranjo PGV, Shojafar M, Mostafaei H, Pooranian Z, Baccarelli E. P-sep: a prolong stable election routing algorithm for energy-limited heterogeneous fog-supported wireless sensor networks. *J Supercomput*. 2017;73(2):733-755.
48. Almutairi AA, El Rahman SA. The impact of IBM cloud solutions on students in Saudi Arabia. Proceedings of the 2016 4th International Japan-Egypt Conference on Electronics, Communications and Computers (JEC-ECC); 2016:115-118; Cairo. doi: 10.1109/JEC-ECC.2016.7518981
49. Roy NR, Chandra P. A note on optimum cluster estimation in LEACH protocol. *IEEE Access*. 2018;6:65690-65696. doi:10.1109/ACCESS.2018.2877704
50. Sarikhani R, Keynia F. Cooperative Spectrum sensing meets machine learning: deep reinforcement learning approach. *IEEE Commun Lett*. 2020;24(7):1459-1462. doi:10.1109/LCOMM.2020.2984430

How to cite this article: Okafor KC, Ndinechi MC, Misra S. Cyber-physical network architecture for data stream provisioning in complex ecosystems. *Trans Emerging Tel Tech*. 2021;e4407. doi: 10.1002/ett.4407