Science, Engineering and Technology Vol. 4, No. 1, pp. 94-111
www.setjournal.com *https://doi.org/10.54327/set2023/v4.i1.107*

Evaluation and Challenges of IoT Simulators for Intelligent Transportation System Applications

Khurram Shehzad Khattak[1], Zawar Hussain Khan[2]
[1]*University of Engineering and Technology Peshawar, Pakistan*
[2]*University of Victoria, BC, Canada*

Abstract

The Internet-of-Things (IoT) constructs a vast, intricate, and perpetually evolving ecosystem exerting profound societal implications. This labyrinthine nature often culminates in errors that directly impact human lives. A significant domain where this complexity materializes is Intelligent Transportation Systems (ITS). Present tools and methodologies inadequately accommodate the complex task of testing and validation, underscoring the urgency for comprehensive review and enhancement. This study aims to present a broad analysis of existing simulators utilized for ITS simulations. It delves into the role and effectiveness of such simulation tools, highlighting their limitations and proposing research directions. This paper scrutinizes both commercial and research-oriented IoT simulators for ITS, evaluating their features and simulation environment tools. We have detailed various ITS scenarios simulated within these frameworks, intending to gauge their readiness for real-world ITS applications and to elaborate on the challenges involved in ITS infrastructure implementation. The findings suggest that despite numerous simulators aiding the evolution of solutions for IoT challenges in recent years, their utility in actual ITS implementations remain uncertain. Consequently, we explore public cloud platforms offering IoT simulation capabilities, focusing particularly on the capabilities provided by the Amazon Web Services (AWS) IoT simulation for this study. Our research outlines the pressing challenges in this field, while proposing potential solutions and flagging opportunities for further research. This study paves the way towards improving the reliability and accuracy of IoT simulators in the context of ITS, which has immense potential to enhance the quality of human life.

Keywords: *Internet of things, IoT simulator, simulation, Cloud computing, Amazon web services.*

1. Introduction

The advent of the Internet of Things (IoT) has paved the way for myriad advancements across various sectors, of which transportation holds a prominent position. Intelligent Transportation Systems (ITS), fortified with IoT technologies, have transformed the landscape of transport, optimizing efficiency and promoting the safe, coordinated, and smart movement of goods and people. ITS are an integral part of smart cities and are continuously evolving with the advent of new technologies [1]. IoT is one such technology that has revolutionized the ITS domain, enabling the seamless communication of devices and applications in real-time. IoT simulators have become an essential tool in this domain, allowing researchers and practitioners to simulate the behavior of IoT devices and evaluate the effectiveness of their solutions before implementing them in the real world. However, alongside the profound benefits it offers, the implementation of IoT in ITS presents a set of significant challenges. In this context, this research aims to explore the evaluation and challenges of IoT simulators for ITS applications.

Corresponding author: *Khurram Shehzad Khattak (khurram.s.khattak@gmail.com)*
Received: *2 October 2023*; Revised: 21 November 2023; Accepted: *29 November 2023*; Published: 15 *December 2023*

Research Context: ITS harness the potential of IoT to facilitate dynamic data interchange among vehicles, infrastructure, and individuals. According to a review on ITS and IoT integration, these systems utilize sophisticated traveler information systems and traffic management systems to significantly enhance transportation operations. They also leverage a broad network of sensors and advanced detection algorithms. Given the intricate and interconnected characteristics of these systems as shown in Figure 1, the utility of IoT simulators is crucial for testing, verifying, and optimizing these systems prior to their real-world implementation.

ITS are designed with an array of services aimed to increase road safety, decrease traffic congestion, and enrich the travel experience for users. Safety measures include alerts about road conditions, while traffic management can involve smart intersections. User experience enhancements may encompass notifications about nearby gas stations, restaurants, and other points of interest. The development and assessment of algorithms and protocols for ITS have been a consistent focus of research. These algorithms and models must be informed by data gathered from existing networks, enabling accurate prediction of the performance of new applications. Assessing such performance is a challenge and typically relies on three distinct methodologies: Mathematical Analysis, Field Operational Tests, and Simulations.

Each method offers its own benefits and limitations, necessitating careful selection as it directly influences the results. Mathematical analysis, for instance, allows for a detailed examination of the issue, yielding valuable insights for better comprehension of the system under study. Statistical distributions are employed to generate models vital for simulation. However, this approach can often oversimplify certain simulation parameters like mobility models, leading to potential inaccuracies in results.

Simulations involve the analysis and study of diverse tools and techniques. A simulator is a specialized system or computational model capable of mimicking real-time events, enabling the evaluation of various hypotheses without exposure to risks.

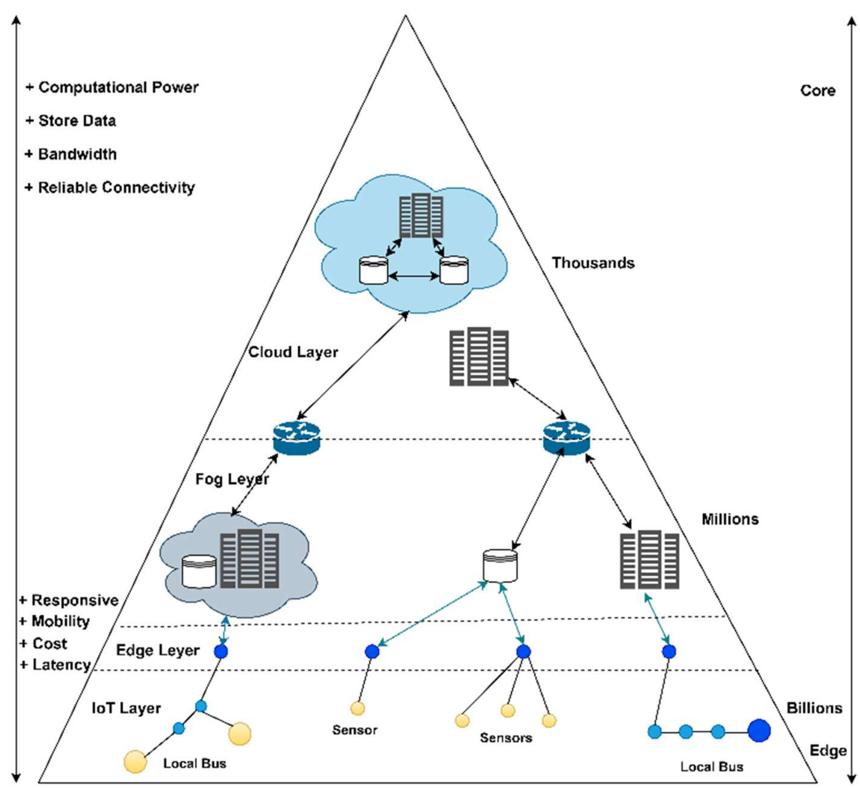

Figure 1 *: Conceptual and architectural model of internet of Things based solutions.*

The IoT simulator aids users in simulating hundreds of user-defined, interconnected devices without creating complications. IoT simulators offer numerous advantages such as control, configurability, manageability, integration, repeatability, and scalability. They enable various devices to communicate seamlessly. IoT device testing and examination is a process of identifying discrepancies between original and intended results. In this context, an IoT simulator proves invaluable in creating a realistic IoT environment to simulate thousands of connected devices seamlessly, as compared to the complex connection of physical IoT devices [2].

Scope and objective: While several comparative and systematic surveys have been conducted in the realm of IoT simulators [2], [3], [4], [5], [6], this research will delve into the multifaceted landscape of IoT simulators and their specific applications within ITS. We will delve into the details of IoT's integration into transportation systems, the current state of IoT simulators, and their role in testing and perfecting ITS applications. Additionally, we will critically evaluate the performance and reliability of different IoT simulators and explore how they handle various challenges, such as managing large-scale settings and integrating machine learning techniques. This research is of critical importance due to the vital role that IoT plays in enhancing ITS. Furthermore, it will illuminate the complexities and challenges in using IoT simulators for ITS applications, contributing valuable insights to the ongoing conversation in this field. As ITS applications become increasingly vital for urban planning and transport optimization, understanding the limitations and potential solutions related to IoT simulators can contribute to their improved functionality and reliability. This research stands to offer guidance for practitioners and policymakers in their efforts to effectively implement and manage intelligent, connected, and optimized transportation systems.

The paper is organized such that section 2 presents an overview of the simulators that have been used to simulate ITS scenarios. It discusses the different simulators and their features, highlighting their relevance to ITS research and development. Section 3 focuses on the challenges associated with these IoT simulators, discussing the limitations and drawbacks that researchers may encounter when utilizing them. This section aims to provide a critical analysis of the current state of IoT simulators and identify areas for improvement. In Section 4, the paper engages in a detailed discussion on the best future research directions in the field of IoT simulators. It explores potential advancements, novel approaches, and emerging technologies that can enhance the capabilities and effectiveness of IoT simulators. Lastly, Section 5 presents the conclusions drawn from the research conducted in this paper, summarizing the key findings and implications for further studies in the field of ITS simulations.

2. IoT Simulators

Simulation plays a critical role in the analysis and study of various tools and techniques, particularly in the context of IoT. Simulators, which are computer programming models or specialized systems, emulate real-time events, allowing researchers to assess the outcomes of hypotheses and assumptions without subjecting them to risks or dangers [7]. By offering control, configurability, manageability, integration, repeatability, and scalability, IoT simulators facilitate continuous communication between multiple devices, thereby providing several benefits.

In the development and testing of IoT applications, simulators assume a vital role, with researchers proposing a range of simulators tailored to IoT use cases. For instance, simulators have been used for simulating different ITS scenarios, enabling developers and researchers to evaluate and optimize ITS applications, particularly in complex scenarios where deploying or accessing physical IoT devices present challenges. These simulators mimic the behavior of mobile and distributed applications using various approaches compatible with different operating systems such as Linux and Windows.

Finding a suitable simulation tool for IoT environments can be challenging, given the limited availability of tools specifically designed for IoT applications. These tools are typically categorized based on their architectural layer and the scope they cover, encompassing areas such as heterogeneous sensors, computation or energy efficiency estimation, scalability, and support for new requirements like protocols. As such IoT simulators can be broadly categorized in three distinct classes namely (1) Full stack, (2) Big data, and (3) Network simulators.

Table 1 : *Comparison of IoT Simulators from ITS Perspective*

	Simulator	Scope / Operational Scale	Type	Application Layers	Language	Built-in IoT Standards	API Integration	Cyber Resilience Simulation	Service Domain	Security Measures
Full Stack Simulator	DPWSim	IoT / Small	Open Source	Application	Java	Secure Web Services Messaging	SOAP	No	Generic	Medium
	iFogSim	Fog / unknown	Discrete event	Perceptual Network\ Application	Java	No	SOAP	No	Generic	Medium
Big Data Processing	Cloudsim	Data Analysis\ Large	Discrete event	Application	Java	Yes	REST	Yes	Cloud Analysis	Yes
	SimIoT	Data Analysis\ Small	Discrete event	Application	Java	No	REST	No	Generic	No
	IoTSim	Data Analysis\ Large	MapReduce Model	Application	Java	No	REST	No	Generic	No
Network Simulators	Cooja	\Small	Discrete Event	Perceptual Network	C\C++	All IoT protocols	REST	Using custom extensions	Generic	Using custom extensions
	QualNet	Discreet Event\ Large		Perceptual Network	C\C++	Zigbee /802.15.4	REST	Yes	Smart City	Yes
	CupCarbon	\ Large	Discrete Event	Perceptual Network	Sen Script	LoRaWAN/ 802.15.4	UDX	No	Smart City	No
	OMNeT++	\ Large	Discrete Event	Perceptual Network	C++	Manual extension	SOAP	Custom Extensions	Generic	Medium
	NS-3	\ Large	Discreet Event	Perceptual Network	C++\ Python	LoRaWAN 802.15.4 6LoWPAN	REST	No	Generic	No
	MobIoTSim	IoT \ Large	Academic	Application Network	C++, C#	Divison Profile for web services (DPWS)	REST	No	Generic	Medium
	EdgeCloudSim	Edge WLAN \ Large	Realistic	Network	Matlab	Mist Computing	SOAP	No	Edge Orchestrator	High
	IoTIFY	Hardware Connection\ Large	Mobile App	Application Network	Python, Java	Real Time	REST	Yes	Smart City	High
	Bevywise-IoT	IoT Device \ Large	Broker	Network	Python, Java	Real Time	REST	No	Smart City	Medium
	Ansys-IoT	IoT industry \ Large	Autonomous	Network	Python, Java	Real Time	REST	YES	Industry	High
	TOSSIM	TinyOS \ Small	Sensor Observation Service	Communication Network	C, Python	Packet injection	REST	Yes	Generic	Medium
	SWE-IoT	WSN	Sensor Observation Service	Communication Network	C, C++	Collision detection	SOAP	No	Human Interface	High

To aid in the selection process, a comparison of top simulators from an ITS perspective has been conducted, with Table 1 presenting a tabulated overview. This comparison examines parameters such as scope, type, programming language, IoT architecture layers, scale of operation, built-in IoT standards, API integration, cyber resilience simulation, service domain, and security measures. Thus providing insights for researchers and practitioners in identifying the most suitable simulator for their specific IoT applications.

2.1. Full stack simulators

Notable examples of specialized IoT simulators in this domain include DPWSim and iFogSim [2], [3], [5]. However, within the context of ITS, a comprehensive review revealed no studies utilizing DPWSim for simulating IoT solutions. Conversely, iFogSim has been selected for eleven IoT solution experimentation initiatives across various ITS scenarios.

iFogSim [2]**,** [3] is an open-source toolkit specifically designed for modeling and simulating resource management techniques in IoT, Edge, and Fog Computing environments. Built upon the widely recognized CloudSim simulator [5], iFogSim provides a comprehensive platform that enables the modeling of intricate systems comprising a substantial number of IoT and Fog nodes. This simulator offers support for sensors, actuators, and application processing elements, facilitating the construction of realistic network topologies and representations of application services. Researchers and developers can leverage iFogSim to assess the performance of various resource management algorithms and policies across diverse scenarios and conditions. Notably, the latest version of iFogSim introduces noteworthy enhancements, including Microservice Management, Dynamic Clustering Mechanisms, and Mobile Management [2]. Additionally, the simulator proves valuable in conducting performance evaluations related to control loop latency, network bandwidth, and energy utilization, particularly when comparing different service deployment approaches, such as cloud-only versus fog computing [5].

Additionally, the simulator proves valuable in conducting performance evaluations related to control loop latency, network bandwidth, and energy utilization, particularly when comparing different service deployment approaches, such as cloud-only versus fog computing [5].

In the context of traffic light optimization, several studies have employed iFogSim simulations to evaluate different approaches. Minh et al. [8] proposed an innovative strategy that harnesses fog computing capabilities to enhance the efficiency of traffic light control systems. By deploying fog computing nodes at intersections, their FogFly system dynamically optimizes traffic light timings based on real-time traffic conditions. The paper provides a comprehensive overview of the design and implementation of FogFly, presenting experimental results that showcase its efficacy in improving traffic flow and reducing congestion. Notably, FogFly achieves significantly lower latency (around 4ms) compared to its cloud-based counterpart (around 200ms). Rampli et al.[9] introduced a solution that leverages fog computing for traffic light optimization. By integrating edge devices and cloud computing, this system adapts traffic light timings based on real-time traffic conditions, aiming to enhance traffic flow and alleviate congestion. The paper presents the detailed design and implementation of this fog-based solution, highlighting its effectiveness in improving traffic management. Through simulation tests, the performance of the fog-based architecture is compared with a traditional TLC-based architecture. The results demonstrate that the fog-based solution achieves significantly lower latency (around 14.74 ms with 30 cars and 15.64 ms with 60 cars) in contrast to the TLC-based architecture (104.43 ms with 30 cars and 108.32 ms with 60 cars), highlighting the potential of the fog-based approach in achieving efficient traffic light optimization. In another study [10], the authors propose an energy-efficient fog computing architecture for smart traffic lights. Their objective is to optimize energy consumption by leveraging fog computing techniques. The paper presents the detailed design and implementation of the proposed architecture, showcasing its effectiveness in reducing energy usage while ensuring reliable communication and data processing capabilities. Through meticulous simulations and experiments, the authors compared their approach with traditional cloud-based architectures, emphasizing the energy-saving advantages of fog computing. The reported findings indicate that fog computing outperforms cloud computing in all scenarios, demonstrating its superior efficiency.

Minh et al. [11] conducted research on efficient task placement in fog computing for IoT application provisioning. Their work addressed the challenge of allocating tasks to fog nodes to minimize latency, energy consumption, and ensure efficient resource utilization. By optimizing task placement strategies, the authors aimed to enhance the performance and scalability of IoT applications. Through simulations, the authors demonstrated the effectiveness of their proposed task placement algorithms, showcasing significant reductions in latency and energy consumption while achieving improved resource utilization.

In another study [12], the authors explored recent advancements and future research directions in the field of edge-cloud-of-things (ECOT). They discussed the challenges and opportunities of ECOT, which combines edge computing, cloud computing, and IoT. The paper presented a comprehensive review of ECOT architectures, technologies, and applications, providing insights into the benefits and potential research areas in this emerging field. A case study was undertaken to illustrate the practical implementation and benefits of ECOT in a real-world scenario. Through iFogSim simulations, the superiority of ECOT is demonstrated based on network usage, latency, and energy consumption.

In [13] , the authors proposed an Emergency Response and Disaster Management System (ERDMS) that utilized fog computing. Their approach prioritizes prompt accident detection and response in real-time. An android application is developed to detect accidents using smartphone sensors, generating an action plan that includes locating the nearest hospital, notifying the emergency department, and contacting the victim's family. Computation tasks are efficiently performed on strategically placed fog nodes in accident-prone areas, minimizing delays and improving responsiveness. The paper provides a detailed description of the system's design and implementation, highlighting its capabilities in detecting accidents and initiating timely responses. Through iFogSim simulations, the authors demonstrate that their fog computing architecture significantly reduces response time, including network latency, throughput, execution time, and average control loop delay, compared to traditional cloud computing approaches.

2.2. Big data simulators

Big data processing simulators play a crucial role in the evaluation and analysis of extensive datasets, with the specific objective of assessing the performance of cloud resources. Prominent simulators in this domain encompass CloudSim, SimIoT, and IoTSim. While these simulators offer specialized capabilities for simulating big data processing, it was observed that SimIoT and IoTSim have not been extensively employed for simulating ITS solutions. In contrast, CloudSim has garnered significant attention and has been employed in seven distinct simulation initiatives that encompass a diverse range of ITS scenarios. This finding highlights the preference and utilization of CloudSim as the simulator of choice for assessing the performance and effectiveness of ITS applications. Further exploration and investigation are warranted to determine the suitability of SimIoT and IoTSim for simulating ITS solutions and to broaden the spectrum of available simulators for comprehensive ITS research and development.

CloudSim is a widely recognized simulator that provides a comprehensive platform for modeling and simulating cloud computing environments. It enables researchers and developers to evaluate the behavior of big data processing applications, resource allocation strategies, and scheduling algorithms in cloud environments.

In [14], [15], [16], various scenarios of vehicular networks were simulated from distinct perspectives. In [14], researchers proposed a technique to enhance the quality of service (QoS) in vehicular ad hoc networks (VANETs) through content dissemination. They introduced a priority-based approach that disseminates content based on its importance, showcasing the effectiveness of their design and implementation in improving content delivery and reducing latency. Through rigorous CloudSim simulations and evaluations, the authors substantiated the advantages of their approach in enhancing the QoS in VANETs, thereby facilitating more efficient and reliable communication among vehicles. In their investigation [15], the authors explored the application of cloud-fog computing in software-defined vehicular networks (SDVNs) and presented a QoS-aware and fault-tolerant approach to enhance their performance and reliability. They provided insights into the design and implementation of their proposed framework, highlighting its dynamic adaptability to changing network conditions and efficient resource allocation. By conducting extensive simulations and evaluations using CloudSim and iFogSim, the authors validated the effectiveness of their approach. The results exhibited noteworthy improvements, including a 50% reduction in response time for safety and non-safety messages by leveraging fog nodes for the SDVN controller, a decrease of up to 4% in execution time, and a significant reduction in the task failure ratio compared to existing models (20%, 15%, 23.3%, and 22.5%). In their study [16] , researchers delved into the design of an efficient hierarchical architecture for SDVNs. They proposed a multi-level architecture that incorporates different hierarchical layers to optimize resource allocation, improve network performance, and enhance scalability. The authors elaborated on the design considerations and implementation details of their proposed architecture. Through CloudSim simulations, they demonstrated the effectiveness of their approach in achieving efficient and reliable communication in SDVNs. Additionally, they introduced a priority-based scheduling algorithm to ensure QoS, prioritizing safety and non-safety messages based on factors such as deadline, size, and constrained location. The simulation results showcased improved performance in terms of execution time for safety messages when compared to non-safety messages.

In [17] , a novel IoT-based traffic violation monitoring system was presented by the authors. Their system demonstrated precise estimation of vehicle speed, passenger count, and seatbelt status across the entire road network. By incorporating edge, fog and cloud computing technologies, the authors achieved high accuracy in detecting traffic violations. Real-life experiments were conducted to compare their system with an alternative approach that solely employs edge and cloud layers, thereby establishing the superiority of their proposed method. While the edge and fog components utilized physical devices, the cloud layer was simulated in CloudSim due to resource limitations. The obtained results revealed enhanced accuracy in passenger counting (97%), seatbelt detection (99%), and reduced root mean square error in speed estimation (1.87), affirming the effectiveness of their IoT-based system in enhancing vehicular safety.

In their work [18], the authors proposed an efficient resource scheduling strategy to optimize the deployment of V2X (Vehicle-to-Everything) microservices in edge servers. They addressed the challenge of resource allocation in edge computing environments by introducing a dynamic resource allocation algorithm that took into consideration service requirements and resource availability. Through simulations conducted using Container-CloudSim, the authors demonstrated the effectiveness of their strategy in improving resource utilization and reducing service response time. Their approach combined a multi-objective model with a multiple fitness genetic algorithm (MFGA) to achieve a balance between resource utilization, resource utilization balancing, and calling distance. The obtained results validated the effectiveness of their MFGA-based strategy in enhancing resource scheduling for V2X microservices in edge servers.

The paper [19] addressed the challenges associated with multimedia content processing in ITS. These systems involved smart vehicles equipped with sensing devices that offer diverse multimedia applications and services. However, the limited computational power and storage capacities of standalone onboard computing devices hinder the efficient and real-time processing of the substantial amount of multimedia data generated by these vehicles. To overcome these challenges, the authors proposed a dynamic priority-based efficient resource allocation and computing architecture for vehicles. In their scheme, multimedia tasks were divided into sub-tasks and assigned to dedicated computing clusters, ensuring timely response delivery based on task priorities. Furthermore, the scheme incorporated dynamic updates of computing resources based on load information. Evaluation using the Cloudsim simulator demonstrated the superiority of the scheme in terms of quality of experience, resource cost, and response time compared to baseline static resource allocation approaches.

In their work, Gilly et al. [20] introduced an end-to-end simulation environment specifically tailored for mobile edge computing (MEC). The authors acknowledged the necessity for a comprehensive simulation tool to effectively evaluate MEC systems. To address this need, they proposed an inclusive simulation environment that integrates various components, including the mobile network, edge servers, and user devices. This environment facilitates the analysis of MEC performance, enabling the assessment of diverse configurations and scenarios. The authors conducted experiments and evaluations to showcase the effectiveness of the simulation environment in assessing MEC system behavior and optimizing its parameters. Additionally, they presented an open-source workflow that integrates well-established simulators like SUMO, OMNeT++, and CloudSim, providing the means to create extensive urban mobile simulation environments.

In their work [21] , the authors concentrated on the design of decentralized control mechanisms for managing large-scale self-adaptive systems. They proposed a reasoning framework called Decor, which supports multi-paradigm modeling, provides a modeling environment for MAPE-K style decentralized control, and offers a co-simulation environment for evaluating the quality attributes of the system. The authors applied Decor to three case studies: an ITS solution, a smart power grid, and a cloud computing application. Through these case studies, they demonstrated the effectiveness of their framework in architecting and evaluating decentralized control in different domains. Their work significantly contributes to the advancement of self-adaptive systems by providing a comprehensive approach for reasoning about and evaluating decentralized control mechanisms.

2.3. Network simulators

The escalating interest in networks WSNs (Wireless Sensor Networks), V2X, VANET, Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I) has precipitated a surge in specialized simulation tools. Selecting an appropriate simulator for WSNs, given the intricate scenarios and myriad protocols it employs, becomes a complex, time-intensive task. The peculiar needs of WSNs, combined with the plethora of available simulators, complicate the selection process. Numerous WSNs simulators have been adapted for IoT environments such as Cooja, QualNet, CupCarbon, OMNeT++, and NS-3, constituting a significant category. These tools, often predating the IoT era, were originally utilized for basic networking or WSN research but have been modified to encompass additional IoT-specific components [5]. In the existing body of literature, comprehensive surveys and comparative analyses related to WSNs, V2X, VANETs, V2V and V2I have been documented [22], [23], [24], [25], [26]. However, in alignment with the specific scope outlined in this study, the focus has been confined to the inclusion of studies and simulators that simulate entire ITS scenarios.

Cooja is a network simulator specifically designed for Contiki, an operating system geared towards low power IoT devices [2], [3] . It has been included in Contiki since version 2.0 and allows for the simulation of both large and small networks of Contiki motes, representing different kinds of sensor nodes within heterogeneous networks. These simulations enable the analysis and control of a functioning Contiki system through the Java Native Interface. The purpose of these simulations is to enhance the understanding and performance of IoT systems.

Hoque et al. [27] proposed an innovative IoTaaS (Internet of Things as a Service) framework for smart cities, utilizing drone-based technology. The framework addresses the challenges of deploying IoT services by introducing a drone-based infrastructure that provides connectivity and computational resources to IoT devices. Through simulations and evaluations, the authors demonstrated the effectiveness of the IoTaaS framework in enhancing performance and scalability in smart cities. They provided a proof-of-concept implementation using RE-Mote IoT devices and the Contiki operating system. The framework's dynamic provisioning capabilities and economic analysis highlight its potential to reduce setup costs and increase the usage of IoT devices. The experiments were conducted using the Cooja simulator, with the IoTaaS service framework running as a web service on a virtual machine hosted in the Amazon Cloud.

Ferreira et al.[28] presented case studies using the Contiki-NG simulator to optimize communication strategies for sensors in an IoT-Fog ecosystem. Their research explored the potential of fog computing in improving sensor performance and efficiency. By analyzing parameters like network topology, transmission power, and routing protocols, they identified effective strategies for sensor communication optimization. The findings offer valuable insights for designing efficient IoT-Fog systems, advancing IoT technologies. Specifically, the study focused on simulating (using Contiki-NG) a hybrid sensor network for public transport in São Carlos. Through performance tests and Multiple Attribute Decision Making (MADM) algorithms, the authors recommended the most suitable deployment location based on multicriteria decision ranking using TOPSIS and VIKOR approaches. This work enables optimal decision-making among various options.

QualNet [2]**,** [3] a network testing and simulation tool developed by Scalable Network Technologies, Inc., imitates the behavior of physical communication networks for planning, testing, and training. Utilizing Parallel Discrete Event Simulation (PDES) algorithms, it processes large-scale, high-fidelity network simulations faster than real-time. QualNet's digital twin technology represents the entire network, including protocol layers, radios, antennas, and devices, providing a platform for performance evaluation and operational issue identification. Its ability to model large networks and their physical layers is unmatched.

In their study [29], the authors focus on enhancing V2X communication in urban areas through a task offloading method. Their dynamic approach takes into consideration the behavior of automobiles in urban environments, leveraging factors such as vehicle velocity and density. This enables the optimization of task allocation between the edge cloud and vehicles. To assess the effectiveness of their method, simulations were conducted using QualNet6.1. The results demonstrated a notable improvement in the performance of V2X communication systems in urban settings. Specifically, the proposed algorithm for edge node selection achieved an average return success of task results that is approximately 3% better than existing methods and 15% better than random selection. In [30] , author explored the scale-free properties of human mobility and their applications to ITS. They investigated the movement patterns of individuals (using QualNet) in urban environments and analyzed the characteristics of their mobility, such as the distribution of travel distances and the frequency of visits to different locations. By uncovering the scale-free nature of human mobility, the authors demonstrated how this knowledge can be leveraged to develop more efficient and effective ITS applications.

The paper [31] presented a spatiotemporal local-remote sensor fusion (ST-LRSF) framework for cooperative vehicle positioning. The authors proposed a method that combined data from local sensors within a vehicle with information from remote sensors in other vehicles to improve the accuracy and reliability of vehicle positioning. By considering both spatial and temporal aspects, the ST-LRSF framework effectively fused the sensor data to estimate vehicle positions in real-time. The performance of the proposed method was evaluated through simulations (SUMO, QualNet) and compared with existing approaches, demonstrating its superiority in terms of accuracy and robustness. In their study, Lee et al. [32] presented an integrated approach that combined a driving hardware-in-the-loop (HIL) simulator with a large-scale VANET simulator to evaluate cooperative ECO-driving systems. The authors aimed to assess the effectiveness and impact of cooperative ECO-driving strategies on fuel consumption and emissions in a realistic driving environment. By integrating the HIL simulator with the VANET simulator (QualNet), they created a comprehensive evaluation platform that considers both the vehicle dynamics and the communication aspects of cooperative ECO-driving. The results of their evaluation provide valuable insights for the development and optimization of cooperative ECO-driving systems in real-world scenarios.

EdgeCloudSim [2] is a unique simulation environment dedicated to edge computing scenarios. It enables researchers to conduct experiments that consider both computational and networking resources. The tool extends the functionalities of CloudSim, a foundational simulation framework, to effectively cater to edge computing contexts. It offers a modular architecture, providing critical features like network modelling for WLAN and WAN, device mobility models, and a tunable load generator. This open-source tool welcomes contributions from the community and is widely used in academic research. Recently, enhancements were proposed to improve its efficiency, scalability, and flexibility, resolving issues related to unnecessary complexity in its original design.

Chawla and Katiyar's study [33] focused on assessing smart and intelligent computing systems within VANETs for time-critical activities. The researchers explored the potential of VANETs in supporting time-sensitive applications, evaluating different computing systems' performance and effectiveness in enabling timely communication and coordination among vehicles. Their assessment provided valuable insights into the strengths and limitations of various computing systems in meeting the requirements of time-critical activities in VANETs. The study also presented a framework for evaluating cloud and edge technologies, comparing criteria like service time, time delay, processing time, and server utilization. The results showed a significant reduction in latency and server usage time (64%) compared to cloud computing, as demonstrated through simulations using EdgeCloudSim and CloudSim technologies.

Kovalenko et al. [34] tackled the challenge of resource allocation in V2I networks using edge computing. Their proposed scheme optimizes resource allocation to enhance the performance and reliability of V2I communications, considering the dynamic nature of vehicular environments and edge computing availability. Through simulations, they demonstrated the effectiveness of their approach in achieving efficient resource utilization and ensuring reliable connectivity in V2I networks. The research also focused on developing a robust V2I system that can handle uncertain request arrivals by dynamically allocating service requests among Base Stations. Simulation results using EdgeCloudSim showed that edge federation can significantly improve the system's robustness by reducing the overall service miss rate by up to 45%.

In [35], authors presented an RSU (Roadside Unit) placement framework for V2I scenarios. The authors focused on determining the optimal locations for RSUs to effectively support V2I communication and services. The framework considered various factors, such as road topology, traffic density, and coverage requirements. By using optimization algorithms and considering both coverage and connectivity aspects, the proposed framework aimed to find the best RSU placement strategy. The effectiveness of the framework was evaluated through EdgeCloudSim simulations.

Belcastro et al. [35] proposed an edge-cloud continuum architecture to address the challenges of urban mobility prediction and planning. They defined three application scenarios involving geotagged data from IoT objects and applied machine learning algorithms for tasks such as next location prediction, location-based advertising, and points of interest recommendation. Their architecture combined edge devices, edge clouds, and central clouds to manage IoT devices and execute machine learning algorithms for data analysis. Through experiments and simulations using the EdgeCloudSim simulator, they evaluated different design choices and demonstrated the improved performance of their approach in terms of processing time, network delay, task failure, and resource utilization.

BevywiseIoT [2] platform is a comprehensive framework for building enterprise-ready IoT applications. The platform offers a reliable solution for mission-critical applications, capable of collecting, visualizing, processing, and analyzing data from IoT devices. It is high-performing and scalable, able to handle millions of devices and high messaging throughput. BevywiseIoT Platform includes components like an MQTT-based message broker for secure device communication. It also supports the development of solutions like Manufacturing Execution System (MES) for Industry 4.0, demonstrating its adaptability and real-world application. Its platform is deployable on-premises or on cloud instances under a user's brand, highlighting its flexibility.

In their work, Lin et al.[36] proposed a bottom-up tree-based storage approach for efficient IoT data analytics in cloud systems. Their proposed storage structure organizes IoT data in a hierarchical tree format, enabling effective retrieval and analysis. By leveraging this tree structure, data can be stored and retrieved based on specific attributes or characteristics, resulting in reduced processing time and computational resources. The approach aimed to optimize IoT data analytics performance in cloud systems, enabling faster and more efficient retrieval and analysis. The effectiveness of their approach was demonstrated through experiments and evaluations, which included using the BevywiseIoT simulator for generating sensing data and NS3 for measuring transmission latency overhead.

TOSSIM [2] is a simulation tool for TinyOS applications. It compiles directly from TinyOS code and can simulate thousands of nodes simultaneously, with every mote in a simulation running the same TinyOS program. TOSSIM provides runtime configurable debugging output and multiple mechanisms for interacting with the network. It also offers a user interface through TinyViz, which allows users to control the simulation, visualize it, inspect debug messages, and interact with the network.

The work by Al-Roubaiey et al. [37] introduced EATDDS, an energy-aware middleware designed for wireless sensor and actuator networks (WSANs). The authors addressed the energy efficiency challenge in WSANs by proposing a middleware that optimized the energy consumption of sensor nodes while ensuring reliable data delivery. EATDDS incorporated energy-awareness mechanisms, such as adaptive sleep scheduling and data aggregation, to prolong the network lifetime and reduce energy consumption. Through TOSSIM simulations, the authors demonstrated the effectiveness of EATDDS in achieving energy efficiency and reliable communication in WSANs. In their work, de Farias et al. [38] introduced a density-based multisensory data fusion approach for multi-application WSNs. Their approach addressed the challenge of efficiently integrating data from multiple sensors by utilizing a fusion technique based on the density of sensor nodes. By considering the spatial distribution and density of sensor nodes, the proposed approach aimed to enhance the accuracy and reliability of data fusion. The authors conducted simulations and experiments to evaluate the performance of their approach, assessing resource consumption and accuracy as key metrics. The results demonstrated the effectiveness of the proposed approach in achieving its goals of improving data fusion quality and resource utilization in multi-application WSNs.

3. Open Challenges

The seamless implementation of the IoT research methodology hinges on the advent of state-of-the-art IoT research tools, specifically those encompassing simulators for proof-of-concept designs and evaluations, and experimental testbeds for tangible prototyping [39] [40]. Ensuring access to, and judicious selection of the optimal research toolkit is an essential precursor for successful outcomes in forthcoming IoT research initiatives.

The following discussion underscores key challenges affiliated with existing IoT simulators which must be rectified in the future. Upon examination of IoT simulators and prevailing research trends, it becomes evident that there exist numerous contributing factors and unresolved research problems [2], [3], [4], [5], [41]. This section puts a spotlight on the research gap present in current IoT simulators in general and ITS in particular.

Rapid Obsolescence: As the landscape of technology and IoT methodologies continues to evolve at an accelerated pace, traditional simulation approaches often fall short in mirroring these advancements. This rapid pace of progression in the IoT realm inherently results in abbreviated product lifecycles, and consequently the swift obsolescence of devices is observed. Such rapid turnover frequently results in an abundance of "orphaned" devices, bereft of vendor support or necessary upgrades. To illustrate, simulators such as iFogSim may face limitations in integrating updated features to reflect emerging advancements in fog computing technologies. The capacity of these simulators to aptly replicate modern communication protocols may also be deficient, thereby prompting rapid obsolescence.

Low Quality Research Datasets: Many traditional simulators, like NS-3 or OMNeT++, lack access to high-quality, real-world datasets for simulation. The lack of such datasets can lead to unrealistic simulation results that don't accurately reflect the performance and behavior of real IoT systems [4] .

Duplication of Effort: Simulators, targeted at various foci such as full-stack, big data, and networking, necessitate researchers and developers to utilize multiple tools for comprehensive IoT environment simulation, resulting in duplicated effort and time wastage. The exceptional heterogeneity of IoT systems, seen in the case studies spanning water treatment facilities to ITS, underscores an intriguing research landscape, but impedes progress due to the absence of a universal design framework and standard communication protocols[4]. Consequently, efforts are duplicated across academia and industry, hampering advancement.

Deficiency in Uniformity: Certain conventional simulators may embody underdeveloped design models, insufficiently addressing the intricate, dispersed characteristics of IoT systems. For instance, simulators such as CloudSim may fail to emulate the intricacy of data management in extensive IoT systems due to their relatively limited design perspectives [42]. Furthermore, the field of IoT simulation suffers from a general absence of consensus regarding optimal practices, primarily due to the multiplicity of simulators in use. This diversity can generate inconsistent results, thus impeding the effective juxtaposition of various IoT methodologies [43]. Moreover, traditional simulators such as Cooja or SimIoT may not adhere to a standardized reference architecture. This lack of a standard framework can complicate accurate modelling of complex IoT systems and pose challenges in ensuring compatibility across different system components [4].

One pertinent issue among the array of features available in IoT simulators is that a single simulator may not possess the capabilities required to carry out a comprehensive investigation of the issues outside its support parameters. Given the concurrent access to infrastructure, the task of identifying hidden, uncertain issues that contribute to suboptimal or unpredictable outcomes becomes considerably complex [44]. Consequently, a crucial research gap is identified: the currently available simulators do not sufficiently support granular investigations aimed at identifying the specific attributes influencing the performance of IoT devices or services [45].

Scalability\Interoperability: The Internet of Things- an expansive network of highly interconnected devices - presents a multidisciplinary domain replete with considerable challenges that prove arduous to fully ascertain. These challenges become particularly prominent during the exploration of scalability and interoperability within the context of IoT. Existing simulation tools provide substantial support for the examination of a myriad of elements including semantic queries, protocols, data transmission, routing, usability, privacy, and security across diverse applications or services. However, these instruments traditionally fall short in addressing scalability complications, which become prevalent during the transformation of problems associated with any factor from intranet-based systems to the internet-wide scope [46]. Both commercially oriented and research-focused simulation tools often fail to identify and consequently, tackle these scalability issues. For instance, conventional simulators like iFogSim or CloudSim demonstrate proficiency in replicating specific scenarios, but grapple with the scalability required for larger, more intricate IoT systems. Such a limitation may hamper their capacity to accurately model and predict the performance of large-scale IoT systems [41].

Interoperability, given the diverse array of devices and protocols incorporated within IoT systems, is a vital factor to consider. Traditional simulators may not extend comprehensive support to this interoperability, thus potentially restricting their effectiveness. Take, for example, the simulator OMNeT++; it might not accurately simulate the interaction between various networking protocols. Despite the common understanding that achieving repeatability of analysis in IoT is quite a challenge, existing simulators frequently fall short in addressing the rising heterogeneity of devices and the associated information. When assessing any use-cases of an IoT system on current simulators, it becomes essential to examine multiple application domains such as smart cities, vehicular IoT, industrial IoT, and so on. Incorporating these elements will significantly enhance the concurrency in accessing the infrastructure, reflecting the extensive nature of the IoT landscape [46].

Mobility: Enhanced device mobility simulations, crucial for smart city and transportation applications, could significantly improve IoT efficiency. Current tools like NS-3, however, lack refined mobility models, undermining simulation precision for such scenarios. An ITS-centric enhancement is necessary for effective IoT simulations.

Concluding our discussion, we see a broad range of open challenges that illuminate the frontier of IoT based ITS research. Given the sheer diversity and volume of IoT devices, an overarching question remains on the strategies that can facilitate the concurrent operation of heterogeneous IoT devices. Furthermore, we face the issue of achieving granularity in IoT investigation, particularly in the context of a dynamic network topology, which comprises an immense number of interconnected devices [46]. A related question involves the development of a cost-effective and versatile simulator that can effectively address local issues and thereby facilitate the comprehensive examination of this expansive technological landscape.

4. Discussion

The amalgamation of our digital and physical terrains is instigating a paradigm shift in the realm of transportation. ITS are progressively integrating methodologies from the IoT, thereby establishing a demand for dependable simulation environments to decipher these complex architectures. A plethora of traditional simulators, such as iFogSim, CloudSim, SimIoT, IoTSim, Cooja, OMNeT++, and NS-3, have been employed, each falling under one of three categories: full-stack, big data, or network simulators. Despite their usefulness, a critical question arises: How will these simulators perform when combined in a cohesive environment, assessing their overall end-to-end performances? This presents a formidable challenge, as these components originate from distinct fields of research.

Large-scale systems and applications, due to the extensive number of devices they rely on, necessitate meticulous analysis and testing before their implementation in target environments due to their inherent complexity. Traditional IoT simulators primarily concentrate on assessing and analyzing low-level networking aspects, with groups of smart objects arranged in particular topologies. The focus isn't on large-scale deployments, which may limit their utility in the broader IoT landscape.

In response to these challenges, academic literature has made efforts towards offering comprehensive solutions. For instance, Kaala, proposed in [47], is a scalable, end-to-end IoT system simulator designed to engage with a variety of real-world IoT cloud services and simulate a broad range of IoT devices and events. This approach mitigates the isolation of IoT experiments from real-world systems, thereby enhancing their accuracy. In another work [48], a novel simulator named DingNet was presented, which was designed to facilitate the exploration and testing of large-scale IoT systems for smart city applications involving mobile devices and stationary gateways. This work addressed the limitations of existing IoT simulators, emphasizing the importance of scalable, adaptable solutions for the rapidly evolving ITS landscape. The paper [49], explored an approach that integrated Eclipse SUMO traffic simulation with the open-source connected vehicle ecosystem Eclipse Kuksa. This approach allowed for a realistic testing of the scalability and functionality of IoT cloud platform architectures for connected vehicles. This overcomes the limited availability of real-world vehicle data and test environments, which have been traditionally restricted due to proprietary constraints. Finally, in the paper [50], a Digital Twin-based tool called TaS was introduced, which aimed to simulate and test IoT environments. The objective here was to refine IoT testing methodologies, predict potential errors, and manage the stresses associated with hardware, software, and physical world aspects. TaS has been tested in an ITS case study, showcasing the close links it forms between the digital twin and the real world.

Overall, these developments highlight the rapid progression in IoT simulation, striving for more robust, accurate, and scalable solutions that can meet the demands of the increasingly complex IoT landscape. Alternatively, a consolidated platform like the Amazon Web Services (AWS) cloud platform is emerging as a potential one-stop solution. This essay intends to compare these two approaches, advocating for the AWS as an all-inclusive IoT simulator for ITS applications [51], [52]. Despite the merits of traditional simulation tools, they share a common drawback - their narrow focus. Simulating an ITS environment holistically would require multiple simulators, a process that may introduce inconsistencies and complexities. Additionally, the steep learning curve associated with mastering these diverse tools could divert resources better allocated to improving the ITS infrastructure itself.

4.1. AWS ITS simulators

In contrast, AWS provides an integrated platform capable of simulating entire ITS ecosystems. AWS IoT services (IoT Device Simulator, Lambda, IoT, storage services to name a few), enable users to create and simulate hundreds of virtual connected devices, model complex ITS infrastructure, analyze intricate data streams, and evaluate the scalability and performance of IoT applications, all within a single platform. Furthermore, it's built on widely used industry standards, reducing the learning curve, and ensuring compatibility with future developments.

The proprietary nature of AWS, however, might pose the risk of vendor lock-in, a risk diversified by the use of multiple traditional simulators. Nevertheless, this risk can be mitigated due to AWS's commitment to open-source projects and robust support for interoperability. AWS provides several unique advantages that make it a desirable choice for simulating IoT applications, including ITS.

4.2. Integration and interoperability

AWS offers IoT Device Simulator, an advanced tool that provides the capability for users to construct and simulate an extensive network of viral connected devices, obviating the requirement for the configuration and management of physical devices. This represents a substantial enhancement over its predecessors, mainly due to its removal of the necessity for both physical devices and labor-intensive scripting operations. The AWS IoT Device Simulator is equipped with an intuitive, web-based graphical user interface (GUI), which significantly simplifies the process of creating simulations. Moreover, the tool is engineered for scalability, with support for up to 1,000 concurrent simulations. This allows users to generate up to 100 simulations at any given instance, thereby facilitating the execution of large-scale testing scenarios

Another crucial aspect of the AWS IoT offering is the usage of AWS Lambda for device simulation. AWS Lambda allows on-demand, serverless compute time, making it easy to simulate a fleet of IoT devices without the need for additional infrastructure. The flexibility of AWS Lambda allows users to specify the number of devices to simulate, the simulation duration, publish frequency, message topic, and data set each time they run a simulation.

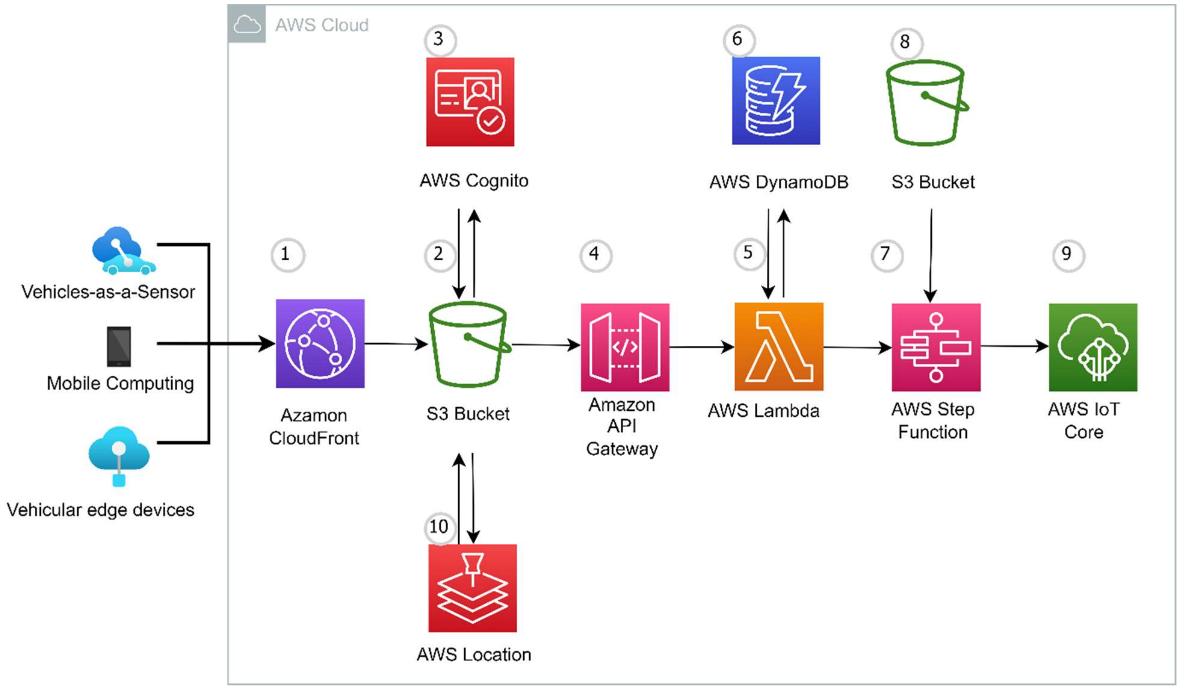

Figure 2. *AWS provided pre-built simulation framework to simulate vehicle telemetry data.*

AWS's interconnected services like AWS IoT Core, AWS IoT Device Management, and AWS IoT Analytics provide a seamless ecosystem for IoT simulation and implementation. This interoperability eliminates the need for multiple, potentially incompatible platforms, streamlining development and testing processes. This point can be illustrated from a AWS provided vehicle telemetry data simulation framework provided in Figure 2 [53].

4.3. Scalability and elasticity

AWS's infrastructure can handle IoT applications at scale. As ITS applications expand, AWS effortlessly scales to meet the growing needs. This flexibility enables realistic stress testing of ITS applications in a range of simulated scenarios.

4.4. Advanced analytics:

AWS provides a holistic solution for big data analytics with its IoT Analytics service, supporting the collection, processing, and analysis of large volumes of data generated from simulated IoT devices. These capabilities are indispensable for refining and enhancing ITS applications.

AWS IoT Analytics is a fully managed service, purpose-built to manage the complexities of IoT data on a petabyte-scale, enabling the handling of substantial volumes of IoT data, processing of messages, and data storage. It comes equipped with pre-built models for common IoT use cases, which simplifies the data analysis process. The service also integrates seamlessly with Amazon QuickSight for data visualization, and with hosted Jupyter notebooks for building, training, and performing machine learning inference. AWS IoT Analytics can filter, transform, enrich, and store device data in a time-series format to facilitate fast retrieval and analysis. Furthermore, AWS IoT Analytics can deal with less structured data, like temperature, motion, or sound data, that might have significant gaps, corrupted messages, or false readings, and require cleaning before analysis.

In addition to IoT Analytics, AWS provides a range of data analytics tools and services, including S3, Kinesis, Glue, Athena, Redshift and Redshift Spectrum, that can be leveraged for data analytics.

4.5. Security

AWS ensures robust security for simulated IoT environments, protecting against potential threats. AWS's security features include encryption in transit with TLS, automated configuration and security checks with AWS IoT Device Defender, and managed authentication and authorization with AWS IoT Core.

4.6. Global presence

AWS's extensive network of regions and availability zones allows for simulating ITS applications under different geographic and network conditions, increasing the accuracy of your simulation scenarios. This global infrastructure ensures a diverse and realistic testing environment for a more robust evaluation of deployed ITS solutions.

4.7. Cost-effectiveness:

AWS's pay-as-you-go pricing model ensures you only pay for the resources you consume, allowing for greater cost control and flexibility. This cost-effective approach enables organizations to optimize their expenses based on actual usage, avoiding upfront investments in excess capacity. Additionally, AWS provides various cost management tools and recommendations to further assist in maximizing the efficiency of your cloud infrastructure spending.

5. Conclusion

In light of the conducted research focusing on the intricate realm of the IoT within the framework of ITS, this study has critically evaluated the existing IoT simulators' proficiency and constraints within this specific domain. Our comprehensive investigation has unveiled a substantial discrepancy between the existing simulation tools and the prerequisites needed for holistic validation and rigorous testing of such systems. This discrepancy, if overlooked, can induce significant inaccuracies.

The examination of the implementation of ITS underscored the necessity for meticulous analysis, bringing to light that both commercially available and research-focused simulators currently do not meet the exacting standards for simulating real-world ITS applications with complete accuracy and unerring reliability. Despite substantial progress in recent years, the conclusion drawn from our research underscores the indeterminate effectiveness of existing IoT simulators in the practical implementation of ITS. The resultant data unequivocally advocates for advancement in simulation tools capable of delivering high-fidelity representations of intricate ITS ecosystems. This claim is further substantiated by our in-depth exploration of diverse ITS scenarios simulated within these frameworks, which unveiled considerable challenges in executing ITS infrastructure.

Simultaneously, our research highlighted the potential of public cloud platforms, such as AWS, in provisioning IoT simulation capabilities. By specifically scrutinizing AWS IoT simulation, our study shed illuminating insights into the capacity of such platforms in ameliorating the encountered challenges intrinsic to ITS.

In summary, the importance of refining the reliability and precision of IoT simulators within the context of ITS has been underscored by this research. Our findings should act as a catalyst for future explorations, spearheading the continuous refinement of tools and methodologies, with the ultimate aspiration of augmenting human existence through the safe and efficient implementation of ITS. Thus, the current research contributes meaningfully to both the field of study and broader societal implications.

Competing Interest Statement

The authors declare no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Data and Materials Accessibility

No additional data or materials were utilized for the research described in the article.

References

[1] S. Ayaz, K. Khattak, Z. Khan, N. Minallah, M. Khan, and A. Khan, “Sensing technologies for traffic flow characterization: From heterogeneous traffic

perspective," *J. Appl. Eng. Sci.*, vol. 20, no. 1, pp. 29–40, 2022, doi: 10.5937/jaes0-32627.

[2] N. D. Patel, B. M. Mehtre, and R. Wankar, "Simulators, Emulators, and Test-beds for Internet of Things: A Comparison," in *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Palladam, India: IEEE, Dec. 2019, pp. 139–145. doi: 10.1109/I-SMAC47947.2019.9032519.

[3] W. Kassab and K. A. Darabkh, "A–Z survey of Internet of Things: Architectures, protocols, applications, recent advances, future directions and recommendations," *J. Netw. Comput. Appl.*, vol. 163, p. 102663, Aug. 2020, doi: 10.1016/j.jnca.2020.102663.

[4] N. Jeffrey, Q. Tan, and J. R. Villar, "Simulators and Testbeds for IIoT Development and Validation," in *2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, Toronto, ON, Canada: IEEE, Jun. 2022, pp. 1–5. doi: 10.1109/IEMTRONICS55184.2022.9795711.

[5] M. Chernyshev, Z. Baig, O. Bello, and S. Zeadally, "Internet of Things (IoT): Research, Simulators, and Testbeds," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1637–1647, Jun. 2018, doi: 10.1109/JIOT.2017.2786639.

[6] A. M. A. Ferreira, L. J. De Melo De Azevedo, J. C. Estrella, and A. C. B. Delbem, "Simulation Tools for the Development of a Public Transport Ecosystem with Mobile Sensors," in *2020 IEEE Cloud Summit*, Harrisburg, PA, USA: IEEE, Oct. 2020, pp. 137–143. doi: 10.1109/IEEECloudSummit48914.2020.00027.

[7] J. P. Dias, F. Couto, A. C. R. Paiva, and H. S. Ferreira, "A Brief Overview of Existing Tools for Testing the Internet-of-Things," in *2018 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, Vasteras: IEEE, Apr. 2018, pp. 104–109. doi: 10.1109/ICSTW.2018.00035.

[8] Q. T. Minh, C. M. Tran, T. A. Le, B. T. Nguyen, T. M. Tran, and R. K. Balan, "FogFly: A Traffic Light Optimization Solution based on Fog Computing," in *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, Singapore Singapore: ACM, Oct. 2018, pp. 1130–1139. doi: 10.1145/3267305.3274169.

[9] M. R. Ramli, R. K. A. Sakir, and D.-S. Kim, "Fog-based Intelligent Transportation System for Traffic Light Optimization: Fog-based Intelligent Transportation System," *Proc. Pak. Acad. Sci. Phys. Comput. Sci.*, vol. 58, no. S, pp. 29–35, Oct. 2021, doi: 10.53560/PPASA(58-sp1)730.

[10] Y. Luo, Y. Chen, and J. Wu, "Energy Efficient Fog Computing with Architecture of Smart Traffic Lights System," in *2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT)*, Surabaya, Indonesia: IEEE, Apr. 2021, pp. 248–254. doi: 10.1109/EIConCIT50028.2021.9431873.

[11] M.-Q. Tran, D. T. Nguyen, V. A. Le, D. H. Nguyen, and T. V. Pham, "Task Placement on Fog Computing Made Efficient for IoT Application Provision," *Wirel. Commun. Mob. Comput.*, vol. 2019, pp. 1–17, Jan. 2019, doi: 10.1155/2019/6215454.

[12] H. Bangui, S. Rakrak, S. Raghay, and B. Buhnova, "Moving to the Edge-Cloud-of-Things: Recent Advances and Future Research Directions," *Electronics*, vol. 7, no. 11, p. 309, Nov. 2018, doi: 10.3390/electronics7110309.

[13] B. K. Dar, M. A. Shah, S. U. Islam, C. Maple, S. Mussadiq, and S. Khan, "Delay-Aware Accident Detection and Response System Using Fog Computing," *IEEE Access*, vol. 7, pp. 70975–70985, 2019, doi: 10.1109/ACCESS.2019.2910862.

[14] M. Ayaz *et al.*, "A Content Dissemination Technique Based on Priority to Improve Quality of Service of Vehicular Ad Hoc Networks," *J. Adv. Transp.*, vol. 2022, pp. 1–16, Apr. 2022, doi: 10.1155/2022/6903450.

[15] S. A. Syed *et al.*, "QoS Aware and Fault Tolerance Based Software-Defined Vehicular Networks Using Cloud-Fog Computing," *Sensors*, vol. 22, no. 1, p. 401, Jan. 2022, doi: 10.3390/s22010401.

[16] M. Adnan *et al.*, "On the Design of Efficient Hierarchic Architecture for Software Defined Vehicular Networks," *Sensors*, vol. 21, no. 4, p. 1400, Feb. 2021, doi: 10.3390/s21041400.

[17] Y.-A. Daraghmi, M. A. Helou, E.-Y. Daraghmi, and W. Abu-ulbeh, "IoT-Based System for Improving Vehicular Safety by Continuous Traffic Violation Monitoring," *Future Internet*, vol. 14, no. 11, p. 319, Nov. 2022, doi: 10.3390/fi14110319.

[18] Y. Shi, Y. Guo, L. Lv, and K. Zhang, "An Efficient Resource Scheduling Strategy for V2X Microservice Deployment in Edge Servers," *Future Internet*, vol. 12, no. 10, p. 172, Oct. 2020, doi: 10.3390/fi12100172.

[19] M. H. Siddiqi, M. Alruwaili, A. Ali, S. F. Haider, F. Ali, and M. Iqbal, "Dynamic Priority-Based Efficient Resource Allocation and Computing Framework for Vehicular Multimedia Cloud Computing," *IEEE Access*, vol. 8, pp. 81080–81089, 2020, doi: 10.1109/ACCESS.2020.2990915.

[20] K. Gilly, C. Bernad, P. J. Roig, S. Alcaraz, and S. Filiposka, "End-to-end simulation environment for mobile edge computing," *Simul. Model. Pract. Theory*, vol. 121, p. 102657, Dec. 2022, doi: 10.1016/j.simpat.2022.102657.

[21] J. Andersson, M. Caporuscio, M. D'Angelo, and A. Napolitano, "Architecting decentralized control in large-scale self-adaptive systems," *Computing*, vol. 105, no. 9,

pp. 1849–1882, Sep. 2023, doi: 10.1007/s00607-023-01167-9.

[22] I. A. Aljabry and G. A. Al-Suhail, “A Survey on Network Simulators for Vehicular Ad-hoc Networks (VANETS),” *Int. J. Comput. Appl.*, vol. Vol 174-No.11, Jan. 2021.

[23] P. Kathuria, P. Adhikari, P. Bhardwaj, F. Khan, R. Nagpal, and D. Mehrotra, “Establishing Inter-vehicle communication using simulation tools,” in *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, Noida, India: IEEE, Jan. 2021, pp. 666–671. doi: 10.1109/Confluence51648.2021.9377130.

[24] M. A. Al-Absi, A. A. Al-Absi, M. Sain, and H. Lee, “Moving Ad Hoc Networks—A Comparative Study,” *Sustainability*, vol. 13, no. 11, p. 6187, May 2021, doi: 10.3390/su13116187.

[25] M. J. Silva, G. I. Silva, F. A. Teixeira, and R. A. Oliveira, “Temporal Evolution of Vehicular Network Simulators: Challenges and Perspectives,” in *Proceedings of the 20th International Conference on Enterprise Information Systems*, Funchal, Madeira, Portugal: SCITEPRESS - Science and Technology Publications, 2018, pp. 51–60. doi: 10.5220/0006696900510060.

[26] B. Shabir, M. A. Khan, A. U. Rahman, A. W. Malik, and A. Wahid, “Congestion Avoidance in Vehicular Networks: A Contemporary Survey,” *IEEE Access*, vol. 7, pp. 173196–173215, 2019, doi: 10.1109/ACCESS.2019.2955142.

[27] M. A. Hoque, M. Hossain, S. Noor, S. M. R. Islam, and R. Hasan, “IoTaaS: Drone-Based Internet of Things as a Service Framework for Smart Cities,” *IEEE Internet Things J.*, vol. 9, no. 14, pp. 12425–12439, Jul. 2022, doi: 10.1109/JIOT.2021.3137362.

[28] A. M. A. Ferreira, L. J. D. M. D. Azevedo, J. C. Estrella, and A. C. B. Delbem, “Case Studies with the Contiki-NG Simulator to Design Strategies for Sensors’ Communication Optimization in an IoT-Fog Ecosystem,” *Sensors*, vol. 23, no. 4, p. 2300, Feb. 2023, doi: 10.3390/s23042300.

[29] H. Matsumoto, B. Gu, and O. Mizuno, “A V2X Task Offloading Method Considering Automobiles’ Behavior in Urban Area,” in *2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Matsue, Japan: IEEE, Sep. 2019, pp. 1–4. doi: 10.23919/APNOMS.2019.8892838.

[30] D. L. Ferreira, B. A. A. Nunes, and K. Obraczka, “Scale-Free Properties of Human Mobility and Applications to Intelligent Transportation Systems,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 11, pp. 3736–3748, Nov. 2018, doi: 10.1109/TITS.2018.2866970.

[31] H.-Y. Jeong, H.-H. Nguyen, and A. Bhawiyuga, “Spatiotemporal Local-Remote Senor Fusion (ST-LRSF) for Cooperative Vehicle Positioning,” *Sensors*, vol. 18, no. 4, p. 1092, Apr. 2018, doi: 10.3390/s18041092.

[32] G. Lee, S. Ha, and J. Jung, “Integrating Driving Hardware-in-the-Loop Simulator with Large-Scale VANET Simulator for Evaluation of Cooperative Eco-Driving System,” *Electronics*, vol. 9, no. 10, p. 1645, Oct. 2020, doi: 10.3390/electronics9101645.

[33] Dr. N. Chawla and Dr. N. Katiyar, “Assessment of Vanet Smart & Intelligent Computing Systems for Time-Critical Activity,” *Online J. Distance Educ. E-Learn.*, vol. Vol 11, Issue 1, Jan. 2023.

[34] A. Kovalenko, R. F. Hussain, O. Semiari, and M. A. Salehi, “Robust Resource Allocation Using Edge Computing for Vehicle to Infrastructure (V2I) Networks,” in *2019 IEEE 3rd International Conference on Fog and Edge Computing (ICFEC)*, Larnaca, Cyprus: IEEE, May 2019, pp. 1–6. doi: 10.1109/CFEC.2019.8733151.

[35] L. Belcastro, F. Marozzo, A. Orsino, D. Talia, and P. Trunfio, “Edge-Cloud Continuum Solutions for Urban Mobility Prediction and Planning,” *IEEE Access*, vol. 11, pp. 38864–38874, 2023, doi: 10.1109/ACCESS.2023.3267471.

[36] J.-W. Lin, J. M. Arul, and J.-T. Kao, “A Bottom-Up Tree Based Storage Approach for Efficient IoT Data Analytics in Cloud Systems,” *J. Grid Comput.*, vol. 19, no. 1, p. 10, Mar. 2021, doi: 10.1007/s10723-021-09553-3.

[37] A. Al-Roubaiey, T. Sheltami, A. Mahmoud, and A. Yasar, “EATDDS: Energy-aware middleware for wireless sensor and actuator networks,” *Future Gener. Comput. Syst.*, vol. 96, pp. 196–206, Jul. 2019, doi: 10.1016/j.future.2019.01.060.

[38] C. Miceli De Farias, L. Pirmez, and F. Coimbra Delicato, “Density Based Multisensor Data Fusion for Multiapplication Wireless Sensor Networks,” in *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech)*, Athens: IEEE, Aug. 2018, pp. 814–821. doi: 10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00-24.

[39] A. Zeb, K. S. Khattak, M. Rehmat Ullah, Z. H. Khan, and T. A. Gulliver, “HetroTraffSim: A Macroscopic Heterogeneous Traffic Flow Simulator for Road Bottlenecks,” *Future Transp.*, vol. 3, no. 1, pp. 368–383, Mar. 2023, doi: 10.3390/futuretransp3010022.

[40] A. Zeb, K. Khattak, M. R. Ullah, Z. H. Khan, and T. A. Gulliver, “HetroTraffSim: A Novel Traffic Simulation Software for Heterogeneous Traffic Flow,” *Sci. Eng.*

Technol., vol. 3, no. 2, Jul. 2023, doi: 10.54327/set2023/v3.i2.78.

[41] G. S. Rapate and N. C. Naveen, "Effectivity Score of Simulation Tools towards Modelling Design in Internet-of-Things," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 5, 2022, doi: 10.14569/IJACSA.2022.0130520.

[42] G. Kecskemeti, G. Casale, D. N. Jha, J. Lyon, and R. Ranjan, "Modelling and Simulation Challenges in Internet of Things," *IEEE Cloud Comput.*, vol. 4, no. 1, pp. 62–69, Jan. 2017, doi: 10.1109/MCC.2017.18.

[43] S. Bosmans, S. Mercelis, J. Denil, and P. Hellinckx, "Challenges of Modeling and Simulating Internet of Things Systems," in *Advances on P2P, Parallel, Grid, Cloud and Internet Computing*, vol. 24, F. Xhafa, F.-Y. Leu, M. Ficco, and C.-T. Yang, Eds., in Lecture Notes on Data Engineering and Communications Technologies, vol. 24, Cham: Springer International Publishing, 2019, pp. 457–466. doi: 10.1007/978-3-030-02607-3_42.

[44] S. Bosmans, S. Mercelis, J. Denil, and P. Hellinckx, "Testing IoT systems using a hybrid simulation based testing approach," *Computing*, vol. 101, no. 7, pp. 857–872, Jul. 2019, doi: 10.1007/s00607-018-0650-5.

[45] R. Casadei, G. Fortino, D. Pianini, W. Russo, C. Savaglio, and M. Viroli, "Modelling and simulation of Opportunistic IoT Services with Aggregate Computing," *Future Gener. Comput. Syst.*, vol. 91, pp. 252–262, Feb. 2019, doi: 10.1016/j.future.2018.09.005.

[46] M. Noura, M. Atiquzzaman, and M. Gaedke, "Interoperability in Internet of Things Infrastructure: Classification, Challenges, and Future Work," in *IoT as a Service*, vol. 246, Y.-B. Lin, D.-J. Deng, I. You, and C.-C. Lin, Eds., in Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 246, Cham: Springer International Publishing, 2018, pp. 11–18. doi: 10.1007/978-3-030-00410-1_2.

[47] U. K. Dayalan, R. A. K. Fezeu, T. J. Salo, and Z.-L. Zhang, "Kaala: scalable, end-to-end, IoT system simulator," in *Proceedings of the ACM SIGCOMM Workshop on Networked Sensing Systems for a Sustainable Society*, Amsterdam Netherlands: ACM, Aug. 2022, pp. 33–38. doi: 10.1145/3538393.3544937.

[48] M. Provoost and D. Weyns, "DingNet: A Simulator for Large-Scale IoT Systems with Mobile Devices," *EWSN Embed. Wirel. Syst. Netw.*, pp. 267–269, Feb. 2019.

[49] P. Heisig, S. E. Jeroschewski, J. Kristan, R. Höttger, A. Banijamali, and S. Sachweh, "Bridging the Gap between SUMO & Kuksa: Using A Traffic Simulator for Testing Cloud-based Connected Vehicle Services," presented at the SUMO User Conference 2019, pp. 213–195. doi: 10.29007/9kkv.

[50] L. Nguyen, M. Segovia, W. Mallouli, E. M. D. Oca, and A. R. Cavalli, "Digital Twin for IoT Environments: A Testing and Simulation Tool," in *Quality of Information and Communications Technology*, vol. 1621, A. Vallecillo, J. Visser, and R. Pérez-Castillo, Eds., in Communications in Computer and Information Science, vol. 1621, Cham: Springer International Publishing, 2022, pp. 205–219. doi: 10.1007/978-3-031-14179-9_14.

[51] A. Zakhil, A. Khan, K. S. Khattak, Z. H. Khan, and A. Qazi, "Vehicular Flow Characterization: An Internet of Video Things-Based Solution," *Pak. J. Eng. Technol.*, vol. 6, no. 1, pp. 13–22, Feb. 2023, doi: 10.51846/vol6iss1pp13-22.

[52] A. Zeb, K. S. Khattak, A. Agha, Z. H. Khan, M. A. J. Sethi, and A. N. Khan, "On-Board Diagnostic (Obd-II) Based Cyber Physical System for Road Bottlenecks Detection," *J. Eng. Sci. Technol.*, vol. Vol. 17 No.2, Apr. 2022.

[53] IoT Device Simulator. [Online]. Available: https://aws.amazon.com/solutions/implementations/iot-device-simulator/