Minimax Theory and its Applications
Volume 11 (2026), No. 1, 81 - 90

Distributed and Scalable Trusted Vehicular System for Secure and Optimized Workflow Execution in Vehicular Edge Computing

Shilpa[1], Dr. Prasanth Thiruvenkadam[2]

[1]Dept. of Computer Science and Engineering, REVA University, Bangalore, Selection Grade Lecturer, Government Polytechnic Immadihalli, Bangalore, Karnataka, India

[2]Associate Professor, School of Computer Science and Engineering, REVA University, Bangalore, India

Abstract
The rapid growth of the Internet of Vehicles (IoV) has created a strong need for low-latency, high-throughput, and secure vehicular edge–cloud (VEC) systems capable of handling large-scale and heterogeneous workloads. However, existing solutions often struggle with limited scalability, rigid trust mechanisms, high computational cost, and weak real-time performance. To address these issues, this work proposes DSTVS-SWE, a distributed and scalable framework for secure workflow execution across vehicular, edge, and cloud layers. The approach models workflows as Directed Acyclic Graphs and combines hierarchical edge–cloud scheduling with dynamic multi-layer trust evaluation using both direct and indirect metrics. The system was implemented in CloudSim and evaluated against the Multi-Agent Deep Deterministic Policy Gradient – Federated Intrusion Detection with Collaborative Cooperative Offloading using Deep Reinforcement Learning (MADDPG-Fed-IDCCO-DRL) benchmark using Montage workloads. Results show notable improvements, including significant reductions in processing time and energy consumption, along with higher throughput and lower latency. Overall, DSTVS-SWE offers a reliable and efficient solution for secure and scalable inter-VEC communication in IoV environments

***Keywords*:** Cloud computing, Edge computing, Internet of Vehicles (IoV), Trust management, Vehicular edge computing (VEC), Workflow scheduling

1. Introduction

The rapid advancement of the Internet of Vehicles (IoV) has significantly reshaped modern Intelligent Transportation Systems (ITS), enabling applications such as real-time data exchange, smart traffic control, autonomous driving, and connected vehicular services. These applications continuously generate large volumes of data that must be processed quickly and transmitted securely across highly dynamic and heterogeneous vehicular networks. To meet these requirements, Vehicular Edge Computing (VEC) has emerged as an effective paradigm that combines the low-latency benefits of edge computing with the extensive processing capabilities of cloud infrastructures [1], [2]. By integrating vehicles, roadside units (RSUs), and cloud servers, VEC supports efficient execution of delay-sensitive applications while reducing network congestion and energy consumption [3], [4]. In recent years, several studies have explored distributed and secure frameworks for improving task scheduling, authentication, and resource management in vehicular environments. Approaches based on vehicle-to-vehicle communication, lightweight cryptographic mechanisms, reputation-based authentication, and blockchain-assisted systems have contributed to enhancing security and data integrity [5]. Similarly, cloud-native orchestration models, federated learning-based scheduling, and UAV-assisted task offloading strategies have demonstrated improvements in data processing and system coordination. Despite these advancements, many existing solutions face challenges related to scalability, latency, dynamic trust management, and seamless integration between edge and cloud layers [6]. A closer examination of current VEC frameworks reveals several limitations. Many approaches rely on static trust models, fixed infrastructure, or pre-configured security mechanisms, which reduce adaptability in highly dynamic vehicular environments [7]. Additionally, the use of computationally intensive techniques such as complex machine learning models or blockchain consensus protocols often introduces processing and communication delays. These factors negatively impact real-time responsiveness and system throughput. Moreover, most existing methods fail to jointly optimize critical performance metrics such as latency, energy consumption, and task execution efficiency across heterogeneous resources, leading to suboptimal performance and reduced reliability [8]. To address these challenges, this work proposes a Distributed and Scalable Trusted

Vehicular System for Secure Workflow Execution (DSTVS-SWE). The proposed framework adopts a hierarchical edge–cloud coordination strategy that dynamically distributes workflows based on workload characteristics, network conditions, and node trustworthiness. Lightweight tasks are executed at the edge to ensure low latency, while computation-intensive tasks are offloaded to cloud virtual machines to maintain scalability. In addition, a dynamic multi-layer trust evaluation mechanism is introduced, incorporating direct, indirect, and historical trust to continuously assess node reliability. This mechanism helps prevent malicious participation, reduce unnecessary retransmissions, and improve overall system performance. By combining trust-aware scheduling with Directed Acyclic Graph (DAG)-based workflow modeling, the proposed approach effectively balances computation and communication overhead, enabling secure and efficient inter-VEC communication.

The key contributions of this work are summarized as follows: The proposed framework dynamically partitions IoV workflows between edge and cloud layers, improving scalability and reducing latency. A multi-layer trust evaluation mechanism integrates direct, indirect, and historical trust to enhance security and reliability. DAG-based workflow modeling combined with hierarchical scheduling optimizes both computation and communication delays. Trust-aware resource allocation ensures efficient utilization of reliable nodes, reducing energy consumption. The overall architecture provides a distributed, scalable, and secure solution for heterogeneous vehicular environments with improved throughput and responsiveness. The remainder of this paper is organized as follows: Section 2 reviews related work, Section 3 presents the proposed methodology, Section 4 discusses the experimental results, and Section 5 concludes the study with future research directions.

2. Literature Survey

Recent advancements in vehicular edge–cloud (VEC) systems have focused on improving task scheduling, security, and communication efficiency in IoV environments. Several distributed frameworks have been proposed to enhance vehicular collaboration and resource utilization. For example, DTOF [1] introduces vehicular edge formation using V2V communication and lightweight hashing, improving efficiency in urban scenarios. Similarly, DP-GHA [2] focuses on attack detection and prevention, enhancing packet delivery and reducing routing overhead. Reputation-based authentication mechanisms, such as the scheme proposed in [3], further improve secure communication through flexible edge selection. In addition, blockchain-based solutions such as BAAIoV [4] and relay-chain protocols [5] provide secure and tamper-resistant data exchange, ensuring trust and transparency. AutoSPADA [6] and V2X-Car Edge Cloud [7] emphasize distributed edge analytics and cloud-native orchestration for improved system scalability. Furthermore, UAV-assisted task offloading and federated learning-based approaches [8] demonstrate the potential of intelligent scheduling and data-driven optimization in IoV environments. Despite these developments, several limitations remain. Many frameworks rely on predefined infrastructure or fixed configurations, limiting adaptability in dynamic vehicular networks, especially under high-density conditions [1], [6]. Approaches based on complex optimization or machine learning models often introduce significant computational overhead and latency, reducing real-time responsiveness [2], [8]. Similarly, blockchain-based systems, while secure, suffer from increased communication and processing delays due to consensus mechanisms [4], [5]. Another critical limitation is the use of static or single-layer trust evaluation mechanisms [3], [9], [10], which fail to capture the dynamic behavior of vehicular nodes. This can result in unreliable node participation, increased retransmissions, and degraded performance. Additionally, several studies focus primarily on either edge or cloud layers without fully integrating both, limiting efficient workload distribution across the edge–cloud continuum [2], [7]. Energy efficiency also remains a concern [12], [13], as many approaches do not jointly optimize latency, throughput, and energy consumption, leading to inefficient resource utilization [14], [15]. To overcome these challenges, the proposed DSTVS-SWE framework introduces a unified and adaptive solution for secure workflow execution. It adopts a hierarchical edge–cloud coordination strategy that dynamically distributes tasks based on workload characteristics, resource availability, and network conditions. Lightweight tasks are processed at the edge to reduce latency, while computation-intensive tasks are offloaded to the cloud to ensure scalability. Furthermore, DSTVS-SWE incorporates a dynamic multi-layer trust evaluation mechanism that combines direct, indirect, and historical trust metrics. This continuous trust assessment enhances node reliability, prevents malicious participation, and reduces retransmissions.

By integrating Directed Acyclic Graph (DAG)-based workflow modeling with hierarchical scheduling, the framework effectively balances computation and communication delays, improving system responsiveness and throughput. In addition, the proposed model emphasizes energy-efficient execution by selecting reliable nodes for task processing, thereby reducing unnecessary energy consumption [15 – 25]. Unlike traditional approaches that rely solely on blockchain or static trust models, DSTVS-SWE provides a distributed, scalable, and trust-aware architecture that integrates security with adaptive workflow scheduling. Overall, the proposed framework addresses key limitations in existing VEC systems by enabling secure, efficient, and scalable workflow execution, making it suitable for real-time and large-scale IoV applications.

3. Methodology

To address the challenges in existing vehicular computation offloading and trust-aware scheduling mechanisms, this study proposes DSTVS-SWE as shown in Figure 1, a distributed and scalable framework for secure workflow execution in heterogeneous vehicular edge–cloud environments. The proposed architecture integrates Internet of Vehicles (IoV), edge computing, and cloud infrastructure into a unified system that improves processing efficiency, minimizes latency, and strengthens workflow security. In this framework, vehicles function as continuous generators of workflow applications, including autonomous driving assistance, real-time video processing, and traffic monitoring. These workflows are initially forwarded to nearby edge servers, typically deployed as Roadside Units (RSUs), which act as the first layer of computation. The edge layer consists of multiple interconnected servers that support localized processing, caching, and quick decision-making. When the edge servers experience resource constraints or when workflows demand higher computational power, tasks are selectively offloaded to cloud-based Virtual Machines (VMs). The cloud layer maintains a diverse pool of VMs with varying capabilities such as processing speed, memory, and bandwidth. Intelligent scheduling strategies allocate tasks dynamically across these resources based on workload requirements, deadlines, and trust levels. This coordinated interaction between vehicles, edge nodes, and cloud resources ensures scalable and energy-efficient execution of large-scale IoV workflows. The edge layer primarily handles delay-sensitive and lightweight tasks to ensure rapid response, whereas the cloud layer processes complex and computation-intensive workflows. This hierarchical design enables efficient task distribution while maintaining system reliability through adaptive trust evaluation.

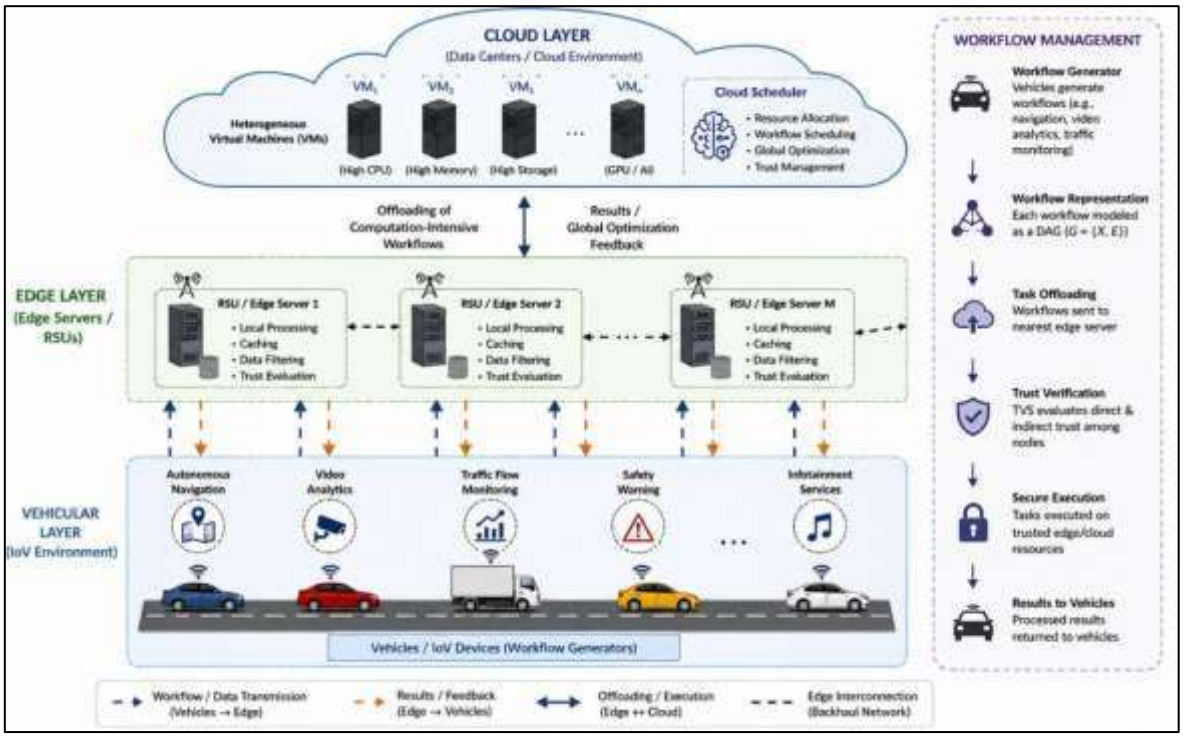

Figure 1. Architecture of the proposed DSTVS-SWE Framework for secure workflow execution in heterogenous vehicular edge-cloud environment.

3.1 IoV Workflow Representation

In the proposed model, each workflow generated within the IoV environment is represented as a Directed Acyclic Graph (DAG), as defined in Eq. (1): Here, $X = \{x_1, x_2, \ldots, x_n\}$denotes the set of individual tasks, while E represents the dependencies among these tasks. Each edge captures the communication cost between dependent tasks executed on different computing nodes. Every task $x_j$is associated with a computational requirement K_j(measured in FLOPs) and is executed on a VM with processing capability F_j. The execution time of a task is expressed as in eq (2): This DAG-based representation allows the scheduler to effectively capture task dependencies and parallelization opportunities, enabling efficient allocation across distributed edge and cloud resources.

$$G = \{X, E\} \tag{1}$$

$$T_j = \frac{K_j}{F_j} \tag{2}$$

3.2 Heterogeneous Vehicular Edge-Cloud Execution Model

The system consists of a set of heterogeneous computing nodes $J = \{1,2,\ldots,N\}$, which include both edge servers and cloud-based VMs. Each node is characterized by its processing capability $f_j$and available communication bandwidth b_j. The overall workflow is divided and distributed across these nodes such that: in eq (3). where G_j represents the portion of the workflow executed on node j. The total execution time U_j (makespan) for each node is given by in eq (4). where E_{comm} represents inter-task data transmission and b_j defines the available bandwidth. This formulation accounts for both computation delay and communication overhead. The optimization objective is to minimize total latency L_{total} and maximize throughput Th, defined as in eq (5) and (6): By jointly considering computation and communication factors, the model ensures balanced workload distribution and improved system responsiveness.

$$\sum_{j=1}^{N} G_j = G \tag{3}$$

$$U_j = \frac{G_j}{f_j} + \frac{E_{comm}}{b_j} \tag{4}$$

$$\text{Minimize: } L_{total} = \max_{j \in J} (U_j) \tag{5}$$

$$\text{Maximize: } Th = \frac{\sum_j G_j}{L_{total}} \tag{6}$$

3.3 Trust-Aware Secure Workflow Execution

Given the dynamic and often unpredictable nature of vehicular networks, ensuring secure and reliable communication is essential. To achieve this, the proposed framework incorporates a Trust Verification System (TVS) that evaluates both direct and indirect trust relationships among nodes. Let xand y represent two interacting nodes, with u representing their interaction duration and o the overall connection period. The security-based trust value is defined as in eq (7): where Sec_{rec} reflects recent interaction behavior and γ is a weighting factor that adjusts the importance of recent observations. Direct trust $D^u_o(x,y)$ is computed as in eq (8): The indirect trust $G^u_o(x,y)$ is derived through intermediate nodes p that have interacted with both xand y in eq (9). where $F^u_o(x,p)$ represents the verified trust-weight between x and p, and Z denotes the set of nodes connected with y. A combined trust score $C^u_o(x,y)$ is then calculated as in eq (10). where δ dynamically adjusts the influence of direct and indirect trust based on interaction patterns. This multi-level trust evaluation helps identify and isolate unreliable or malicious nodes, ensuring secure workflow execution and data integrity across the network.

$$Sec^u_o(x,y) = \gamma\, Sec_{rec}(x,y) + (1-\gamma)\, Sec^u_{o-1}(x,y) \tag{7}$$

$$D^u_o(x,y) = Sec^u_o(x,y) \tag{8}$$

$$G_o^u(x,y) = \frac{\sum_{p \in Z-\{x\}} F_o^u(x,p) \times D_o^u(p,y)}{\sum_{p \in Z-\{x\}} F_o^u(x,p)} \quad (9)$$

$$C_o^u(x,y) = \delta D_o^u(x,y) + (1-\delta) G_o^u(x,y) \quad (10)$$

3.4 Performance Objectives Tradeoffs Optimization

The proposed DSTVS-SWE model establishes a strong relationship between execution time, energy consumption, latency, and throughput. Since energy consumption is proportional to execution time ($E = P \times T$) and throughput is inversely related to time ($Th = D/T$), minimizing execution time becomes a key objective. By incorporating trust-aware scheduling, the system ensures that only reliable nodes are selected for task execution. This reduces the likelihood of task failures, retransmissions, and unnecessary reallocations, ultimately improving overall throughput. Furthermore, adaptive distribution of workflows between edge and cloud resources minimizes communication delays and supports system scalability. The integration of trust evaluation with intelligent scheduling enhances both performance and security. In summary, the DSTVS-SWE framework provides: reduced latency through localized edge processing, improved throughput through parallel and secure execution, scalable resource utilization through dynamic VM allocation, and enhanced reliability through continuous trust assessment. This approach establishes a robust foundation for secure, efficient, and scalable workflow execution in dynamic vehicular edge–cloud environments.

4. Results And Discussion

This section evaluates the effectiveness of the proposed DSTVS-SWE framework in enabling secure and scalable workflow execution within inter-vehicular edge–cloud (inter-VEC) environments. The performance is analyzed using multiple metrics, including processing time, energy consumption, throughput, and latency. The experimental setup is first described, followed by a detailed analysis of each performance indicator.

4.1 System Configuration

The proposed model was implemented using the CloudSim-based [16] simulation platform namely CloudSimSDN [17] to emulate a realistic vehicular edge–cloud environment. The trust computing validation is employed through IoTSim-Osmosis package [18] within CloudSimSDN. The experiments were conducted on a system running Windows 11, equipped with an Intel i9 (12th Generation) processor, 32 GB RAM, and a CUDA-enabled GPU with 4 GB memory. The implementation was carried out primarily in Java, with additional support from C#. The simulated infrastructure consists of two data centers, each configured with four Physical Machines (PMs) and a total of eight Virtual Machines (VMs). Each PM is provisioned with 32 GB RAM, 1 TB storage, and 1000 Mbps bandwidth, while each VM is assigned 32 GB storage and 5 Mbps bandwidth. To assess system robustness under adversarial conditions, a Denial-of-Service (DoS) attack scenario was incorporated. For comparative evaluation, the MADDPG-Fed-IDCCO-DRL model [8] was implemented under the same configuration. The CIC dataset [19] was used to simulate IoV-related cyberattacks, while the Montage workload dataset [20] was utilized to represent computational tasks of varying complexity.

4.2 Processing Time

The comparison of processing time reveals that DSTVS-SWE consistently outperforms the benchmark model across all workload sizes as shown in Figure 2. For smaller workloads (25 tasks), the proposed method completes execution in 28 ms, significantly lower than the 120 ms required by the baseline approach. As the number of tasks increases, the performance gap becomes even more pronounced. For instance, at 1000 tasks, DSTVS-SWE completes execution in 950 ms, whereas the benchmark requires 8100 ms. This substantial reduction in processing time can be attributed to the hierarchical task distribution strategy and DAG-based scheduling. By intelligently assigning lightweight tasks to edge nodes and offloading computation-intensive operations to cloud VMs, the system minimizes processing bottlenecks and enables efficient parallel execution.

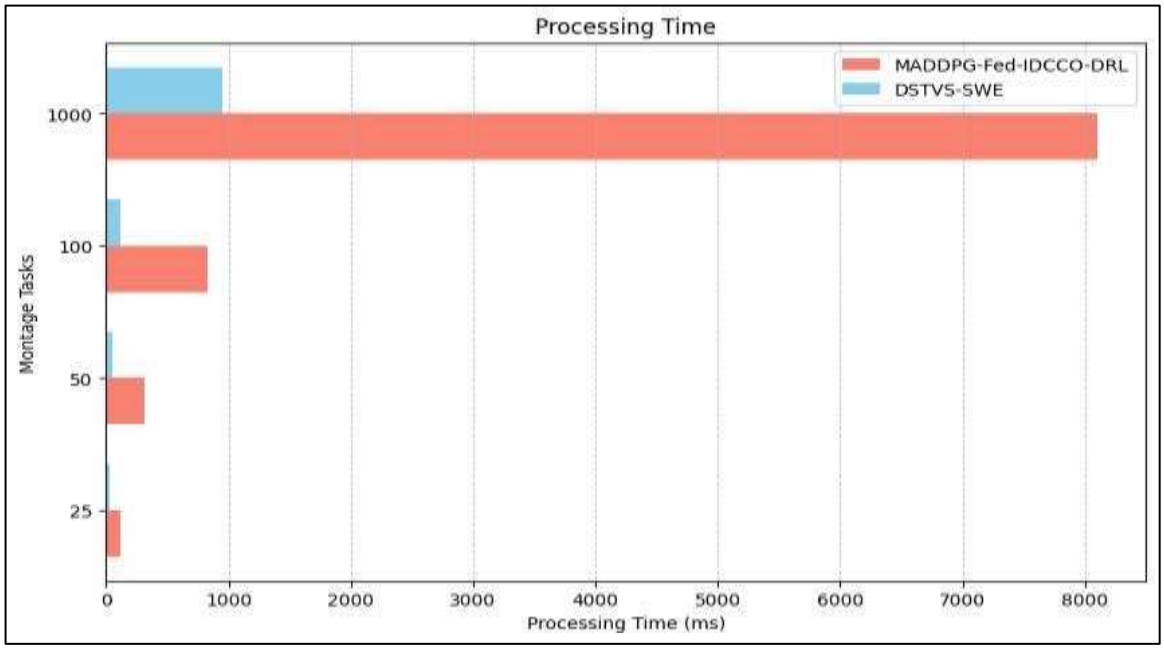

Figure 2. Processing Time.

4.3 Energy Consumption

The energy consumption analysis shows that DSTVS-SWE achieves notable reductions across all workload levels as shown in Figure 3. For smaller workloads, energy usage drops significantly, and this trend continues as task volume increases. At the largest workload (1000 tasks), the proposed model consumes only a fraction of the energy required by the benchmark system. The primary reason behind this improvement lies in the relationship between execution time and energy consumption. By reducing task completion time, the system directly lowers overall energy usage. Additionally, the trust-aware scheduling mechanism ensures that only reliable nodes participate in execution, thereby avoiding unnecessary retransmissions and reducing power wastage across the network.

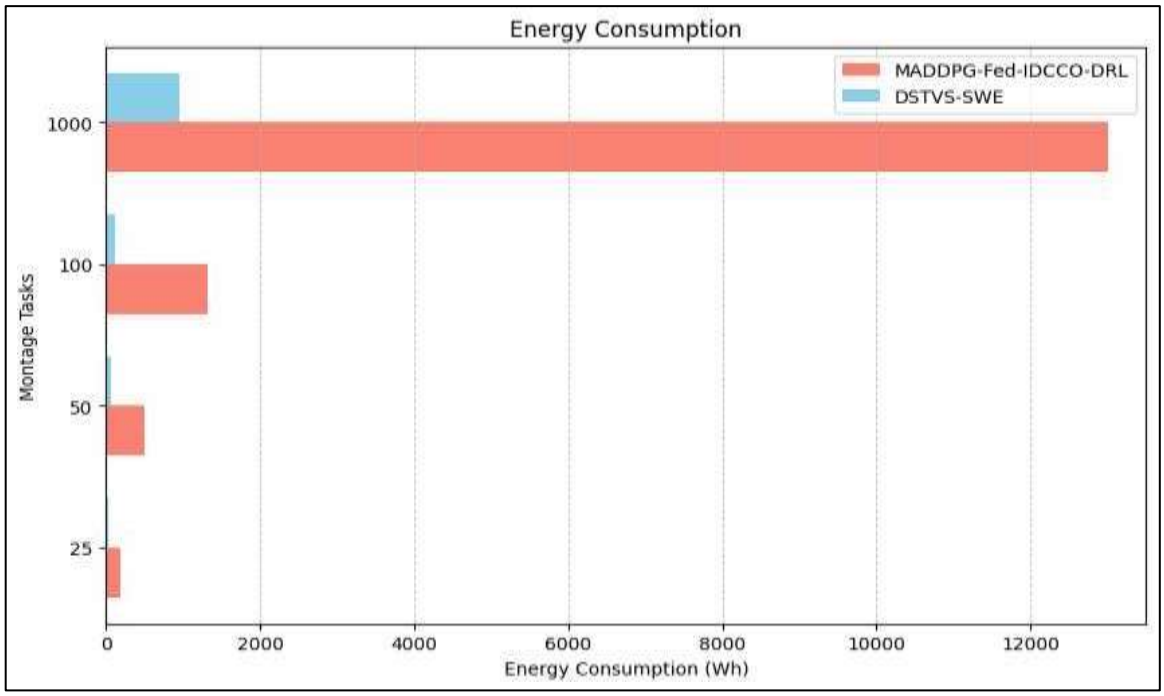

Figure 3. Energy Consumption.

4.4 Throughput

Throughput analysis demonstrates that DSTVS-SWE achieves a significantly higher task processing rate compared to the benchmark model as shown in Figure 4. Even under increasing workload

conditions, the proposed system maintains stable and high throughput levels. This improvement is driven by efficient resource allocation and dynamic workload balancing between edge and cloud layers. By reducing idle time and enabling concurrent task execution, the system ensures continuous data flow. Moreover, the integration of trust-based filtering minimizes disruptions caused by unreliable nodes, further enhancing effective throughput.

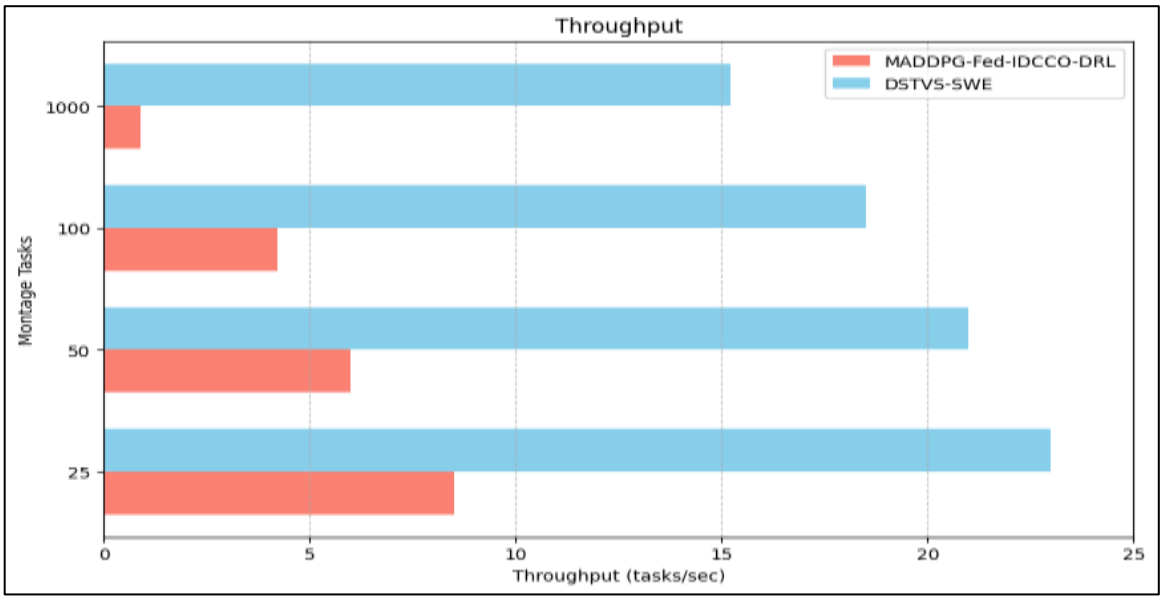

Figure 4. Throughput.

4.5 Latency

Latency measurements indicate that DSTVS-SWE provides consistently lower response times per task as shown in Figure 5. The system effectively reduces delays across all workload scenarios, including large-scale task execution. The reduction in latency is primarily due to the edge–cloud coordination strategy. Time-sensitive tasks are processed at the edge layer, close to the data source, while computationally intensive tasks are offloaded to the cloud. This balanced approach minimizes both computation delay and communication overhead. Additionally, trust-based node selection reduces network instability, further contributing to lower latency.

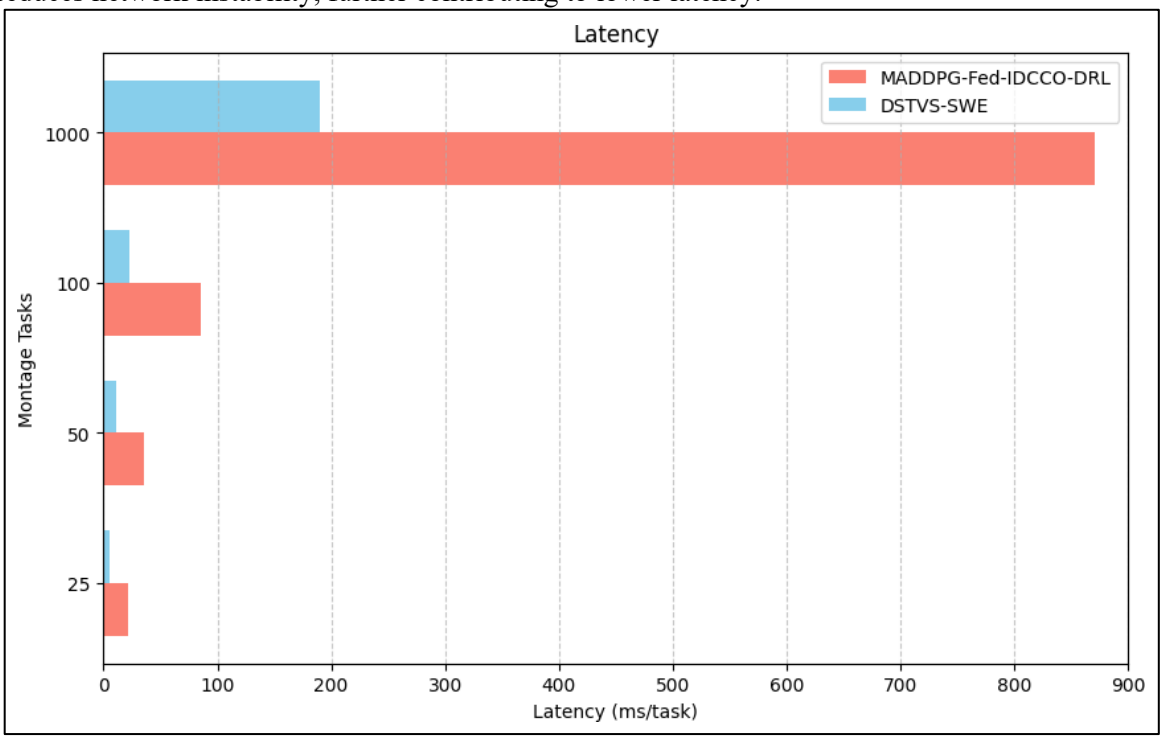

Figure 5. Latency.

4.6 Discussion and Limitation

The overall results clearly presented in Table 1, indicate that DSTVS-SWE delivers significant improvements across all evaluated performance metrics. The model demonstrates strong scalability, maintaining stable performance even as workload size increases. The combined use of hierarchical scheduling, adaptive resource allocation, and trust-aware decision-making enables efficient workflow execution in dynamic vehicular environments. One of the key strengths of the proposed approach is its ability to simultaneously optimize performance and security. The trust evaluation mechanism effectively filters out unreliable or malicious nodes, reducing failed executions and retransmissions. This not only enhances system reliability but also indirectly improves energy efficiency and latency. Compared to traditional approaches, particularly those relying heavily on reinforcement learning or blockchain-based consensus mechanisms, DSTVS-SWE achieves faster response times with lower computational overhead. The absence of complex consensus delays allows the system to operate in near real-time, making it well-suited for latency-sensitive IoV applications. However, despite these advantages, certain limitations remain. The effectiveness of the model depends on the availability of consistent trust information and stable communication links. In highly dynamic vehicular scenarios, where frequent disconnections or sparse node density occur, the trust evaluation process may be temporarily affected. This can lead to reduced scheduling accuracy and minor performance degradation. Although the framework incorporates adaptive mechanisms to handle such conditions, extreme mobility or unstable network environments may still pose challenges. Addressing this limitation requires further enhancements, such as predictive trust modeling and decentralized caching strategies, to ensure uninterrupted operation even under highly volatile conditions.

Table 1. Performance Comparison of DSTVS-SWE and MADDPG-Fed-IDCCO-DRL.

Metric	Montage 25	Montage 50	Montage 100	Montage 1000	Average Improvement (%)
Processing Time (ms)	76.7 % ↓	82.3 % ↓	85.8 % ↓	88.3 % ↓	**83.30%**
Energy Consumption (Wh)	83.8 % ↓	88.5 % ↓	91.1 % ↓	92.7 % ↓	**89.00%**
Throughput (Tasks/s)	170.5 % ↑	250.0 % ↑	340.0 % ↑	1588.0 % ↑	**587.10%**
Latency (ms/task)	77.3 % ↓	69.4 % ↓	72.9 % ↓	78.2 % ↓	**74.50%**

5. Conclusion

This study presented DSTVS-SWE as a practical approach to improve performance, scalability, and security in vehicular edge–cloud environments. The motivation stems from the growing complexity of IoV systems, where high mobility, diverse nodes, and continuous data generation make it challenging to maintain low latency, high throughput, and secure communication. A review of existing methods revealed key limitations, including poor scalability, rigid trust mechanisms, high computational overhead, and limited real-time responsiveness. To overcome these challenges, DSTVS-SWE was designed using a hierarchical edge–cloud coordination strategy combined with dynamic multi-layer trust evaluation and energy-aware scheduling. The model was implemented in a CloudSim-based environment and evaluated against the MADDPG-Fed-IDCCO-DRL benchmark under varying workloads. The results showed clear improvements, including significant reductions in processing time and energy consumption, along with noticeable gains in throughput and latency performance. Overall, the framework provides a balanced solution that improves both efficiency and security. Future work will focus on predictive trust modeling, decentralized caching, and real-world validation using 5G-enabled vehicular systems to further enhance adaptability and reliability.

Input Image Size	224 × 224
Gradient Clipping	1.0

ACKNOWLEDGEMENT
I would like to express our sincere gratitude to all those who have supported and contributed to this research project. Primarily, I extend our heartfelt thanks to our guide for his unwavering guidance, invaluable insights, and encouragement throughout the research process. No funding is raised for this research.

FUNDING INFORMATION: No funding is raised for this research.

AUTHOR CONTRIBUTIONS STATEMENT:

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Shilpa	✓	✓	✓		✓	✓		✓	✓	✓	✓		✓	
Dr. Prasanth Thiruvenkadam2	✓	✓		✓	✓	✓		✓	✓	✓	✓	✓		

C : **C**onceptualization M : **M**ethodology So : **So**ftware Va : **Va**lidation Fo : **Fo**rmal analysis	I : **I**nvestigation R : **R**esources D : **D**ata Curation O : Writing - **O**riginal Draft E : Writing - Review & **E**diting	Vi : **Vi**sualization Su : **Su**pervision P : **P**roject administration Fu : **Fu**nding acquisition

CONFLICT OF INTEREST: The Author declares no conflict of interest.

DATA AVAILABILITY: Datasets utilized in this research in reference [20].

References

1. S. Mittal, R. K. Dudeja, R. S. Bali, and G. S. Aujla, “A distributed task orchestration scheme in collaborative vehicular cloud edge networks,” Computing, vol. 106, no. 4, pp. 1151–1175, Oct. 2022, doi: 10.1007/s00607-022-01119-9.
2. A. Malik, M. Z. Khan, S. M. Qaisar, M. Faisal and G. Mehmood, "An Efficient Approach for the Detection and Prevention of Gray-Hole Attacks in VANETs," in IEEE Access, vol. 11, pp. 46691-46706, May. 2023, doi: 10.1109/ACCESS.2023.3274650.
3. J. Zhang, H. Zhong, J. Cui, L. Wei and L. Liu, "CVAR: Distributed and Extensible Cross-Region Vehicle Authentication With Reputation for VANETs," in IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 1, pp. 74-89, Jan. 2024, doi: 10.1109/TITS.2023.3306547.
4. A. Alsaeed, S. Almowuena and A. N. Alyahya, "BAAIoV: A Blockchain-Based Authentication and Authorization Framework for Secure and Reliable Internet of Vehicles Communication," in IEEE Access, vol. 13, pp. 150821-150837, Aug. 2025, doi: 10.1109/ACCESS.2025.3601003.
5. X. Tang et al., "A Verifiable Privacy-Preserving Cross-Chain Protocol for Trusted Vehicle Edge Computing," in IEEE Transactions on Vehicular Technology, Oct. 2025, doi: 10.1109/TVT.2025.3625439.
6. A. Nilsson, S. Smith, J. Hagmar, Magnus Önnheim, and Mats Jirstrand, “The AutoSPADA platform: User-friendly edge computing for distributed learning and data analytics in connected vehicles,” Internet of Things, vol. 30, pp. 101480–101480, Dec. 2024, doi: 10.1016/j.iot.2024.101480.
7. D. Ku, H. Zang, A. Yusupov, S. Park, and J. Kim, “Vehicle-to-Everything-Car Edge Cloud Management with Development, Security, and Operations Automation Framework,” Electronics, vol. 14, no. 3, pp. 478–478, Jan. 2025, doi: 10.3390/electronics14030478.
8. X. Wang, C. He, W. Jiang, W. Wang and X. Liu, "Generative AI-Based Dependency-Aware Task Offloading and Resource Allocation for UAV-Assisted IoV," in IEEE Open Journal of the Communications Society, vol. 6, pp. 3932-3949, 2025, doi: 10.1109/OJCOMS.2025.3562720.

9. I. S. Alkhalifa and A. S. Almogren, "Enhancing Security and Scalability in Vehicular Networks: A Bayesian DAG Blockchain Approach With Edge-Assisted RSU," in IEEE Access, vol. 12, pp. 116558-116571, July. 2024, doi: 10.1109/ACCESS.2024.3429184.
10. U. Tariq and Tariq Ahamed Ahanger, “Enhancing Intelligent Transport Systems Through Decentralized Security Frameworks in Vehicle-to-Everything Networks,” World Electric Vehicle Journal, vol. 16, no. 1, pp. 24–24, Jan. 2025, doi: 10.3390/wevj16010024.
11. A. Borah and A. Paranjothi, “Enhancing VANET Security: An unsupervised learning approach for mitigating false information attacks in VANETs,” Electronics, vol. 14, no. 1, p. 58, Dec. 2024, doi: 10.3390/electronics14010058.
12. C. Li, K. Xiao, J. Xu, W. Ji and L. Gao, "Game-Theoretic Optimization for Task Offloading and Resource Allocation in Parked Vehicle-Enhanced Internet of Vehicles," in IEEE Internet of Things Journal, doi: 10.1109/JIOT.2026.3668937.
13. B. Lin, Q. Chen, X. Chen, W. -K. Jia, Y. Lu and N. N. Xiong, "DQPS: An Intelligent Multi-Hop Computation Offloading Scheme for Workflow Applications in Vehicular Edge Computing Networks," in IEEE Transactions on Consumer Electronics, vol. 71, no. 1, pp. 2202-2216, Feb. 2025, doi: 10.1109/TCE.2024.3502408.
14. Z. Li, J. Gong, X. Xiong and D. Wang, "Multi-Slot Secure Offloading and Resource Management in VEC Networks: A Deep Reinforcement Learning-Based Method," in IEEE Access, vol. 13, pp. 4533-4546, 2025, doi: 10.1109/ACCESS.2024.3524636.
15. Y. Yin et al., "Mobility-Aware Assisted Deep Reinforcement Learning for Collaborative Task Migration and Resource Allocation in Vehicular Edge Computing," in IEEE Transactions on Vehicular Technology, doi: 10.1109/TVT.2026.3660321.
16. R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose, and R. Buyya, “CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms,” Software: Practice and Experience, vol. 41, no. 1, pp. 23–50, Jan. 2011.
17. Jungmin Son, Amir Vahid Dastjerdi, Rodrigo N. Calheiros, Xiaohui Ji, Young Yoon, and Rajkumar Buyya, “CloudSimSDN: Modeling and simulation of software-defined cloud data centers,” in Proceedings of the 15th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing (CCGrid), 2015, pp. 475–484, doi: 10.1109/CCGrid.2015.8.
18. K. Alwasel et al., “IoTSim-Osmosis: A framework for modeling and simulating IoT applications over an edge-cloud continuum,” Journal of Systems Architecture, vol. 116, p. 101956, Jun. 2021, doi: 10.1016/j.sysarc.2020.101956.
19. “Datasets | Research | Canadian Institute for Cybersecurity | UNB,” www.unb.ca. https://www.unb.ca/cic/datasets/index.html.
20. “https://pegasus.isi.edu/workflow_gallery/gallery/montage/index.php,” Pegasus WMS. https://pegasus.isi.edu/workflow_gallery/gallery/montage/index.php (Accessed 21. Oct 2025).
21. A. Waheed et al., "A Comprehensive Review of Computing Paradigms, Enabling Computation Offloading and Task Execution in Vehicular Networks," in IEEE Access, vol. 10, pp. 3580-3600, 2022, doi: 10.1109/ACCESS.2021.3138219.
22. J. Lee and W. Na, "A Survey on Vehicular Edge Computing Architectures," 2022 13th International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, Republic of, 2022, pp. 2198-2200, doi: 10.1109/ICTC55196.2022.9952556.
23. Y. Dai and Y. Zhang, "Adaptive Digital Twin for Vehicular Edge Computing and Networks," in Journal of Communications and Information Networks, vol. 7, no. 1, pp. 48-59, March 2022, doi: 10.23919/JCIN.2022.9745481.
24. W. Qi, X. Xia, H. Wang and Y. Xing, "A Task Partitioning and Offloading Scheme in Vehicular Edge Computing Networks," 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), Norman, OK, USA, 2021, pp. 1-5, doi: 10.1109/VTC2021-Fall52928.2021.9625369.
25. J. Zhang, H. Guo, J. Liu and Y. Zhang, "Task Offloading in Vehicular Edge Computing Networks: A Load-Balancing Solution," in IEEE Transactions on Vehicular Technology, vol. 69, no. 2, pp. 2092-2104, Feb. 2020, doi: 10.1109/TVT.2019.2959410.