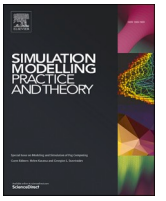
Simulation Modelling Practice and Theory 120 (2022) 102621

Contents lists available at ScienceDirect

Simulation Modelling Practice and Theory

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



A secure drone-to-drone communication and software defined drone network-enabled traffic monitoring system

Adarsh Kumar [a], Anuraj Singh Yadav [a], Sukhpal Singh Gill [b,*], Haris Pervaiz [c], Qiang Ni [c], Rajkumar Buyya [d]

[a] School of Computer Science, University of Petroleum and Energy Studies, Dehradun, India
[b] School of Electronic Engineering and Computer Science, Queen Mary University of London, UK
[c] School of Computing and Communications, Lancaster University, United Kingdom
[d] Cloud Computing and Distributed Systems (CLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Australia

ARTICLE INFO

Keywords:
Security analysis
Lightweight security mechanism
Distributed computing
Message passing interface (MPI)
Unmanned autonomous vehicles (UAV)

ABSTRACT

This paper proposes a novel lightweight security-enabled distributed software-defined drone network (SDDN) for traffic monitoring. Security of drone/Unmanned Aerial Vehicles (UAV) communication and data exchange is ensured through lightweight key generation and encryption/decryption algorithm. A hybrid (static and dynamic) OpenMP/MPI-based distributed processing is used to compute the security primitives for drone-to-drone communication. The proposed approach is more reliable, scalable and interoperable compared to other centralized logical control and incorporating network programming methods. Additionally, the use of cryptographic primitives and protocols make it more secure against attacks. A comparative analysis of proposed lightweight key generation and encryption/decryption algorithms with state-of-the-art algorithms shows that both proposed algorithms require fewer Gate Equivalents (GEs), and it varies from 18.4k to 29.6k. In terms of performance, both algorithms' computational delay varies from 1.5 to 2 s. Jitter lies between 0.7 msec and 2 msec. The proposed algorithms are found to have communicational costs varying with 0.4 and 0.7 times of input in bytes with a base value of 1.4 and 1.25. Further, energy consumption is varying with 0.4 and 0.7 times of input in bytes with a base value of 0.3 and 0.25. Security interruption probability variation analysis show that the proposed security algorithms are better compared to state-of-the-art approaches. Further, security analysis of both algorithms (using a statistical and formal model) shows that the proposed system is protected against various attacks.

1. Introduction

Drones are widely accepted in day-to-day applications such as product delivery, surveillance, monitoring, search and rescue, and military operations. Applications like traffic engineering, road safety, and on-road incident monitoring require multi-drone cooperative movement and collision-free strategy for drones. This provides efficiency, flexibility, reliability, and lesser costs for real-time

* Corresponding author.
E-mail addresses: adarsh.kumar@ddn.upes.ac.in (A. Kumar), anurajs@ddn.upes.ac.in (A.S. Yadav), s.s.gill@qmul.ac.uk (S.S. Gill), h.b.pervaiz@lancaster.ac.uk (H. Pervaiz), q.ni@lancaster.ac.uk (Q. Ni), rbuyya@unimelb.edu.au (R. Buyya).

https://doi.org/10.1016/j.simpat.2022.102621
Received 2 December 2021; Received in revised form 15 June 2022; Accepted 22 June 2022
Available online 25 June 2022
1569-190X/

traffic monitoring [1–3]. In any drone-based application, the major issues that need to be addressed so far include placement of drones, drones movement and path planning, increasing drone's payload capacity, efficient and effective drone's resource management and optimizing the drone's flying, secure communications among flying drones with lightweight cryptographic primitives and protocols.

The major challenges in the traditional traffic engineering system include [1–3]: (i) The fixed sensor, camera and display system based traffic system do not cover all regions. Thus, the domain of area coverage and availability of information is limited to certain road area only, (ii) Vehicle data captured in the traditional system is hard to apply a data fusion approach. This is because tracing the target vehicle over the road is almost impossible [4,5]. In a drone-based system, it would be much easier to trace any target vehicle even if it moves off the road, (iii) The processing of data using drones and sharing with controller or cloud resources would be much efficient, faster, and secure compare to the traditional system, (iv) The drone-based moveable monitoring system can easily replace drones if it malfunctions without affecting the on-road traffic. Whereas, the traditional system causes much unease, (v) A single or small number of drones movement and LiDAR/Optical/Radar-based collision avoidance strategy for traffic engineering can cover certain areas only. A pre-planned drone movement strategy allows a significant number of drones to constantly cover a wide area in stipulated time [5,6]. As a result, pre-planned multiple drone-based on-road traffic monitoring will be very useful especially over highways/motorways, and (vi) Traffic engineering equipment attached with a drone-based moveable system reduces the vulnerabilities as drones will be randomly deployed and the exchange of information should be through secure and lightweight cryptographic primitives and protocols. Additionally, Software Defined Networking (SDN)-integrated approach will be useful through various means. The importance of SDN-integrated approach is summarised as follows [7]:

- In a SDN-integrated connected vehicular network, the possibility for increased speed and agility in the availability of network devices (both virtual and real) to users as a result of decreasing the requirement for human involvement [4]. Growing use of virtual private networks (VLANs) as a component of physical local area networks (LANs) has created a tangled web of dependencies and links between them. Using virtual and physical networks, SDN abstracts the control and data planes, making it easier to centralise corporate management and provisioning in vehicular networks.
- Administrators may experiment with network configuration without having to worry about creating an interruption in service to the network when utilising SDN-integrated vehicular monitoring system [5]. Administrators may manage both physical and virtual switches and network devices from a single centralised control point, which is very handy. This central point can effectively manage the monitoring activities of vehicular network. SDN provides a single set of APIs. These APIs can be used to construct a

Table 1
Comparative analysis of SDN-based approach with other approaches in vehicular network or its infrastructure.

Parameter	SDN or SDDN-based Vehicular Infrastructure	Traditional Network-based Vehicular Infrastructure	Cloud/Edge/Fog Computing-based Vehicular Infrastructure (without virtualization)	Cloud/Edge/Fog Computing-based Vehicular Infrastructure (with virtualization)	Network Functions Virtualization (NFV)
Flexibility and Efficiency in Resource Allocation	High	Low	Medium	High	High
Costs	Low	High	High	Low	Low
Functions	Separate Network control and Forwarding functions. Data and control plane mounted over different decoupled by software	Network control and forwarding is same but differentiate control from data plane. However, data and control plane mounted over same plane	Network control, data and forwarding functions can be customized as per requirements	Network control, data and forwarding functions can be customized as per requirements	Create Network abstract from hardware
Programmable	Network is programmable	Network is not programmmable	Network is programmable	Network is programmable	NFV support SDN in programmable infrastructure
Control	Centralized	Distributed	Distributed	Distributed	Centralized or Distributed
Interface	Open interface	Close interface	Open Cloud Computing Interface (OCCI) is available	Open Cloud Computing Interface (OCCI) is available	Can adopt open interface from SDN
Prioritization and Blocking Packets	Both prioritization and blocking can be performed	No prioritization or blockchin packets. All packets are same.	Network can be customized for packet prioritization or blocking.	Network can be customized for packet prioritization or blocking.	Network can be customized for packet prioritization or blocking.
Structural Complexity	Low	High	High	Low	Low
Extensibility	High	Low	Low	High	High
Troubleshoort and Reporting	Easy	Difficult	Difficult	Easy	Easy

single management console that can be used to manage both physical and virtual devices. Thus, a holistic approach is available to effectively manage the network required to monitor, control and operate the vehicular network and associated infrastructure.

- It is possible that security and policy information will be delivered and configured consistently across the vehicular infrastructure as a result of the adoption of the SDN Controller [6]. Since the introduction of virtualization, network management has grown more challenging in vehicular networks as well. While it is conceivable that centralising security management under a single authority may aid vehicular network in managing security more effectively, doing so in a safe and acceptable way must be taken into consideration.
- SDN may be used to combine and automate numerous network administration tasks in order to save costs [8]. Additionally, more efficient servers, and improved virtualization management contribute to lower overall expenses.
- Commodity hardware optimization might be made simpler with SDN. With appropriate and maximum SDN controller instructions, the hardware used in vehicular network and infrastructure management may be reused and more cost-effective equipment can be added. SDN may be used more effectively if it is integrated with existing network infrastructure [9]. Thus, no pre-planning or new model implementation is required to improve the services in the vehicular network.
- SDN can be integrated with cloud, fog or edge networks supporting services to vehicular networks and give an abstract view of data centres constituted and maintained by hardware devices in these networks [10]. Thus, SDN provides effective network and security management of vehicular infrastructure through cloud-based support.
- SDN can provide flawless services to vehicular network users and service providers. It ensures Quality of Service (QoS) in text or multimedia data transmission in vehicular communications [11]. It adds cooperative sensing to vehicular networks which in turn reduces the cost of information sharing in vehicular networks. It improves the QoS by providing an abstraction of the heterogeneity of vehicular networks. Heterogeneity in wireless networks and communication provides cost-effective solutions in information exchange [12].

Table 1 shows the comparative analysis of SDN-based approach with other approaches.

In this work, the usefulness of the SDN network is explored by simulating drone and SDN integrated environments for vehicle communication. To analyze the performance of SDN with vehicular networks, QoS parameters (delay and jitter) are evaluated. This paper addresses the problem of surveillance and traffic monitoring system. Here, a UAV-based traffic monitoring automated system with monitoring and surveillance capabilities is proposed. With the integration of the SDN-based approach, flexibility, efficient resource utilization (in terms of gate equivalents), connectivity, efficient network management, security and interoperability if drone network and vehicle communication is possible and ensured in this work. The proposed system integrates SDN-based system to monitor and control the drone-devices. The proposed system provides drone-to-drone collision detection and avoidance strategy. This strategy integrates a predetermined multi-layered drone-movement approach for developing collision-free environment. Further, drone-to-drone communication is secured using lightweight key-generation and encryption/decryption algorithms. Security analysis of proposed lightweight security algorithms is performed statistically and formally using ProVerif toolkit. Finally, the performance of a drone-based traffic monitoring system is measured using simulators.

The major contributions of our paper are as follows:

- We presented multi-layered drone movement with collision avoidance having distributed software-defined drone network (SDDN) support to monitor and direct the individual drone's functionalities.
- We proposed and evaluate the lightweight drone secure communications and data transfer algorithms. These algorithms ensure the secure key exchange and security of data in its transmission, processing, and storage stages.
- The proposed security mechanisms are verified using a formal mathematical model and ProVerif toolkit.
- We measured the performance of security algorithms and drone-networks. Simulations are designed to evaluate and analyze the performances.
- We implemented Message Passing Interface (MPI) and OpenMP-based parallel and distributed computing system for hash function computation in security algorithms used in traffic engineering equipment and on-road traffic monitoring.

The rest of the paper is organised as follows. Section 2 summarises related work on drone-based traffic monitoring systems. This section also presents symbols and notations used in this paper. Section 3 presents the proposed cloud and SDDN-based architecture for traffic engineering and monitoring systems. Section 4 shows the drone-movement, traffic data collection and collision avoidance algorithms. Section 5 presents the proposed lightweight drone-to-drone secure data transfer algorithms. Section 6 performs the statistical security analysis of proposed algorithms. The simulation, and performance analysis of the proposed system is depicted in Section 7. Section 8 summarises and concludes the paper.

2. Related work

The necessity of monitoring on-road vehicular traffic has been investigated in many studies [3,13–20]. Altshuler et al. [18] proposed swarms of reconnaissance drones-based vehicles monitoring systems capable of on-demand and cost-effective optimal area coverage strategies considering a given roads networks. In [3,13,14], similar approaches are explored where single or multiple drones are programmed to monitor and surveillance road traffic, incidents, traffic volume count, gather traffic information, and time-varying vehicular flows. Garcia-Aunon et al. [17] presented an aerial swarm-based traffic monitoring approach simulated in Unity game engine. This approach controls the aerial swarm using six behaviors parameters and optimizes the parameters using genetic algorithm.

The proposed approach shows a good performance (25% improvement in efficiency for Swarm City with 0.5% standard deviation) with use of 23 parameters. The proposed scheme uses image-based surveillance system where swarms are forced to visit new places without returning to old ones. Further, a two-layer surveillance system is proposed. In this system, pheromones are produces to monitor the traffic zones. The quantity of pheromones is dependent over on-road traffic conditions. It increases or decreases either with pre-defined equations or using number of cars measurements. This paper has majorly concentrated over improving the system efficiency for a given Swarm City. However, the chances of collisions with drone movement from one location to another cannot be neglected in proposed approach. To detect and avoid collisions, drone-to-drone communication is important to consider especially in traffic monitoring system. Further, security aspects in drone-to-drone communication are required to be addresses.

Christodoulou et al. [21] proposed a drone-based traffic monitoring across a particular region with pre-defined monitoring points. The real road network topologies are demonstrated for the proposed approach that minimize the travel time while keeping the resource constraints into consideration. In this paper, two drone-based monitoring schemes (cheapest insertion algorithm (CIA) and multiple tour algorithms (MTA)) are analyzed. CIA is a weighted graph-based greedy heuristic algorithm that allows the drone to traverse through those pre-identified points that minimizes the overall cost. In MTA, cluster-based mapping is done to identify the route that minimizes the drone's travel cost. After testing the proposed system over three different networks, it has been realized that MTA outperforms the CIA with random initial node placement. This experimentation confirms that the pre-identified routes and pre-planned drone-based monitoring over those routes can give collision free environment with minimum traveling cost. Khan et al. [22] experimented the on-road vehicle's speed measurement using UAV-based system in Saudi Arabia. The proposed UAV-based system overcomes the limitations of the SAHER system and it helps in decreasing the number of deaths and injuries. The UAV-based smart surveillance system with 5G technology helps in faster traffic violation cases detection. In [22], three-layer architecture is proposed to distinguish the functionalities of on-road traffic, communications using telemetry devices and 5G, and drone-monitoring system. All these functionalities are proposed to be controlled through a base station. The experimentation results of proposed approach implementation show that it reduces the number of accidental incidents, and helps the people in following the rules and regulations.

Balasubramanian et al. [23] introduced the Local Traffic-Aware Green Algorithm based on Sleep-Scheduling (LTGAS), which considers traffic patterns while scheduling sleep in autonomous networks. This approach is built on the Sleep-Scheduling algorithm. During periods of heavy congestion, the LTGAS system considers online and local traffic statistics to turn off some underutilised network nodes and connections and turn on certain sleep devices, as appropriate. Due to the small magnitude of the impact of this proposal on network performance measures such as end-to-end latency, packet delivery ratio, and average link utilisation, it is not

Table 2
Comparison analysis of drone-based traffic monitoring systems.

Author	Year	Application	A	B	C	D	E	F	G	H	I	J	K	L	M	N
Lee et al. [3]	2015	Drone-based traffic and incident monitoring	✓	✓	✓	x	✓	x	x	x	✓	x	x	x	x	x
Shi et al. [13]	2018	Drone-based multi-purpose monitoring and surveillance system	✓	x	✓	x	✓	✓	✓	✓	✓	x	x	x	x	✓
Khan et al. [14]	2017	Drone-based flight planning, traffic monitoring, analysis and vehicle trajectory analysis	✓	✓	✓	x	✓	x	x	x	✓	x	x	x	x	x
De-Bruin et al. [15]	2015	Drone-based traffic profiling and flow estimation	✓	✓	✓	x	x	x	✓	x	✓	x	x	x	x	x
Niu et al. [16]	2018	Drone-based traffic profiling and data analysis	✓	✓	✓	x	✓	x	x	x	✓	x	x	x	x	x
Garcia-Aunon et al. [17]	2019	Swarms of drones for traffic monitoring and surveillance	✓	✓	✓	x	✓	x	x	x	x	✓	✓	x	✓	✓
Altshuler et al. [18]	2018	Drone-based pre-defined region monitoring for target manoeuvring in a road-network scenario	✓	✓	x	x	✓	x	x	✓	✓	x	x	x	x	✓
Barmpounakis [26]	2020	A drone-based traffic monitoring system (named pNEUMA) for urban area	✓	✓	✓	x	x	x	x	x	✓	x	x	x	✓	✓
Congress et al. [27]	2020	Drone-based vehicle redirection for hazardous obstruction induction using image processing	✓	✓	✓	x	x	x	x	x	✓	x	x	x	✓	✓
Hamurcu and Tamer [28]	2021	Selection and raking of drones for traffic management	x	✓	x	x	✓	x	x	x	x	x	x	x	✓	x
Guirado et al. [29]	2021	Drone-based simulation is performed for monitoring on-road traffic conditions	✓	✓	x	x	✓	x	x	x	x	x	x	✓	✓	✓
Kumar and Jain [30]	2021	Drones for monitoring ground objects.	x	✓	✓	x	✓	x	x	x	x	x	x	x	✓	x
Basu et al. [31]	2022	Drone and Softwarized 5G-based dynamic resource sharing and network management	x	x	✓	✓	✓	x	x	x	✓	x	x	x	x	✓
Butilă and Boboc [32]	2022	Survey of Drone-based traffic monitoring and analysis	✓	✓	✓	x	✓	x	x	x	x	x	x	x	x	x
Proposed System	2022	Drone-based traffic profiling and monitoring system with secure key and data exchange and hash computation using parallel and distributed computing process.	✓	✓	✓	✓	✓	✓	✓	✓	x	✓	✓	✓	✓	✓

A: Traffic monitoring, B: Traffic data collection, C: Image processing, D: Parallel & distributed computing, E: Drone-usage, F: Drone-to-drone collision detection, G: Drone-to-drone collision avoidance, H: LiDAR/Optical/Radar-based collision detection & avoidance, I: Layer-free drone movement, J: Single-layer drone movement, K: Multi-layer drone movement, L: Inductive loop data, M: Drone direction observation, N: Drone speed observation.

negatively affected by this proposal. Additionally to the variables just stated, there may be additional factors that have an impact on the energy conservation efficiency of other network components. The only methods that have been shown to reduce the energy consumption of a network component are those that do not rely on the functioning of other network components to achieve this reduction. A new dynamic method can be developed based on the findings of this study, in which specific network nodes and connections are dynamically turned off during low traffic periods to conserve energy, while selected sleep devices are dynamically turned on during high traffic periods as needed. The proposed approach can be extended for computational and communicational cost analysis in addition to network performance and energy conservation evaluations. Further, security analysis of proposed approach can be taken up for analysis in future. Nguyen et al. [24] proposed a traffic monitoring framework named as DeepMonitor. DeepMonitor is a new IoT traffic monitoring system that is built on SDN technology. In addition, it has the capability of doing fine-grained analysis of IoT traffic at the network edge. When used in combination with an intrusion detection system, DeepMonitor has been shown to improve the detection of distributed denial-of-service (DDoS) attacks. As a consequence, it is possible that the proposed approach will be able to optimise the average long-term granularity degree for all traffic kinds. Its goal is to get the optimum policy for a Markov Decision Process (MDP) system as quickly as possible without the requirement for prior knowledge of the traffic behaviour of IoT devices. While maintaining the best policy, a federated reinforcement learning-based IoT traffic monitoring system reduces the amount of DDQN training time required while also decreasing total training time. In particular, they show that using the optimal flow rule match-field control policy increases traffic granularity while decreasing flow-table overflow, resulting in improved DDoS attack detection performance as a result of the policy. This work can be extended to include analysis of proposed system against various other cyber-attacks.

Kyrkou et al. [25] proposed deep-learning enabled system for analyzing emergency situations like flood, fire, on-road accidents, fire or building collapsed. In this approach, the emergency aerial image classification method is used. Using this method, a UAV device can generate an emergency response with high performance than existing models with minimum on-device memory requirements. In this

Table 3
Comparison analysis of data handling in traffic monitoring systems.

Author	Year	Application	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Cucchiara et al. [1]	2000	Image processing and rule-based intelligent traffic monitoring system	✓	✓	✓	✓	x	x	x	x	x	x	x	✓	x	✓	x
Liu et al. [2]	2017	Bus rider's mobile data-based traffic monitoring system	✓	✓	✓	✓	x	x	x	x	✓	✓	x	✓	x	✓	x
Niu et al. [16]	2018	Drone-based traffic profiling and data analysis	✓	✓	✓	x	x	x	x	x	✓	x	x	x	x	x	x
Garcia-Aunon et al. [17]	2019	Swarms of drones for traffic monitoring and surveillance	✓	✓	✓	✓	x	x	x	x	✓	x	x	x	x	x	x
Altshuler et al. [18]	2018	Drone-based pre-defined region monitoring for target manoeuvring in a road-network scenario	✓	✓	✓	x	x	x	x	x	✓	x	x	x	x	x	x
Barmpounakis [26]	2020	A drone-based traffic monitoring system (named pNEUMA) for urban area	✓	✓	✓	x	x	x	x	x	✓	x	x	x	x	x	x
Congress et al. [27]	2020	Drone-based case studies for T-section and rail-crossing on-road vehicle data collection system	✓	✓	✓	x	x	x	x	x	✓	✓	✓	x	x	✓	x
Gia et al. [33]	2020	LoRa, Edge and Fog computing-based vehicle tracking and traffic monitoring	✓	✓	✓	✓	x	✓	✓	x	✓	✓	x	x	✓	x	x
Beg et al. [34]	2019	UAV-based vehicle identification, traffic monitoring, data collection and profiling system	✓	✓	x	✓	x	x	x	✓	✓	✓	✓	x	x	✓	x
Balasubramanian et al. [23]	2020	An energy-efficient traffic-aware green algorithm based on Sleep-scheduling is proposed for autonomous networks	✓	✓	✓	✓	x	x	x	x	✓	✓	x	✓	x	✓	x
Balasubramanian et al. [35]	2022	Edge Intelligence in the Internet of Vehicles for traffic monitoring	✓	✓	✓	✓	x	x	x	✓	✓	✓		✓	✓	✓	✓
Bathla et al. [36]	2022	Autonomous vehicles and need of intelligence automation for traffic management and vehicular networks	✓	✓	✓	x	x	x	x	✓	✓	x	x	x	x	x	✓
Verma et al. [37]	2022	Smart traffic management system and veicular network profiling	✓	✓	✓	✓	x	x	x	✓	✓	✓	x	x	✓	✓	x
Srikanth and Kumar [38]	2022	Vehicle number plate reading and traffic profiling	✓	✓	✓	✓	x	x	x	x	✓	✓	x	✓	x	x	x
Proposed System	2022	Drone-based traffic profiling, data collection and monitoring system with secure key and data exchange, and hash computation using parallel and distributed computing process	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	x	✓	✓	✓	✓

A: Data collection, B: Data Processing, C: Data analytics, D: Data visualization, E: Private MPI-based Task distribution, F: Private MPI-based task execution, G: OpenMP/Cloud-based task storage, H: Distributed resource allocation for traffic data handling, I: Traffic profiling, J: Vehicle profiling, K: Weather information measurement (wind, moisture etc.) for drone trajectory, L: Data error measurements, M: Vehicle-length & model-based detection, N: Vehicle speed measurement, O: Data security.

paper, a dataset is formulated that collect the images and grouped them together according to incidents. A drone-based data collection and the trained system are proposed with improved performance-accuracy tradeoffs. Likewise, many UAV-based solutions are proposed for traffic monitoring [15–18]. In [15], importance is drawn over to use the drones for traffic flow estimation and tracking. Traffic flow data is useful for future planning and preparing the performance metrics that can reduce the overall cost of traffic monitoring system deployment. In [16–18], video and data processing processes are integrated in one system to analyze the vehicles, its type, speed and traffic flows. Further, a web-application is designed and developed to display the statistics remotely. As a large amount of data is generated in drone-based traffic monitoring system, there is a lack of proposal to integrate the security aspects which include security proposal for (i) data storage at drone-device and remote server, (ii) data processing at drone-device, and an intermediate or remote server, and (iii) data communication/exchange between drones.

Table 2 shows a comparison of the proposed system with the state-of-the-art drone-based traffic monitoring systems. Table 3 illustrates a comparison analysis of data handling approaches in traffic monitoring systems.

In the literature, various SDN-based approaches are discussed to improve IoT network performances [39,40]. For example, Rabet et al. [41] discussed the use of SDN for IoT networks and proposed SDMob i.e. SDN-based mobility management. The SDMob design is reliant not only on an external controller but also on an Internet of Things network that has certain limitations. This is essential for it to be able to deliver the lightweight mobility management capabilities that it provides since this is required for it to do so. In terms of performance, SDMob is better than both RPL and ARMOR, despite the fact that both of them contain a large amount of overhead. SDMob's ability to more efficiently spread messages is the basis of its advantage over its rivals in the marketplace. It is very clear that this is the issue that we are dealing with when we examine the ratio of the total length of time to the number of packets that were sent. Because SDMob is being used in the network, it is now possible to design a network architecture that is in a position to provide location precision down to the decimeter level. This feat is made feasible due to the fact that the network makes use of the SDMob platform. In the past, there was not even a distant possibility to act in such a manner. Because of this, even when there are only a few mobile nodes involved, the network is still able to come very close to reaching a delivery rate of one hundred percent for the packets that it delivers. This is because mobile nodes are able to communicate with each other even when there is a significant distance between them. Mobile nodes are able to interact with one another even when there is a substantial distance between them. Even if there is a significant distance between them, mobile nodes are still able to communicate with one another. Balasubramanian et al. [35] discussed new applications for the Internet of Vehicles are finding it more difficult to send considerable amounts of data wirelessly due to the fact that wireless networks are becoming less dependable and more crowded. These data can include the location of the vehicle as well as the information gathered by various sensors. Because wireless networks will surely get more crowded and less trustworthy as time goes on, this shift will need to take place at some point in the near future. Before this transfer can take place, there is a chance that a considerable amount of travel will be necessary on the part of the parties involved. In order to effectively manage both forms of traffic, it is essential to have a complete understanding of both the traffic on data networks and the traffic on roads. If we are going to be successful in conquering the problems that we have been discussing up to this point, this is something that we really need to pay attention to. Within the confines of this investigation, we put up a different instructional strategy as a potential fit for the hybrid VeNet learning system. This approach was created in order to take use of the geographical and temporal correlations that are present in

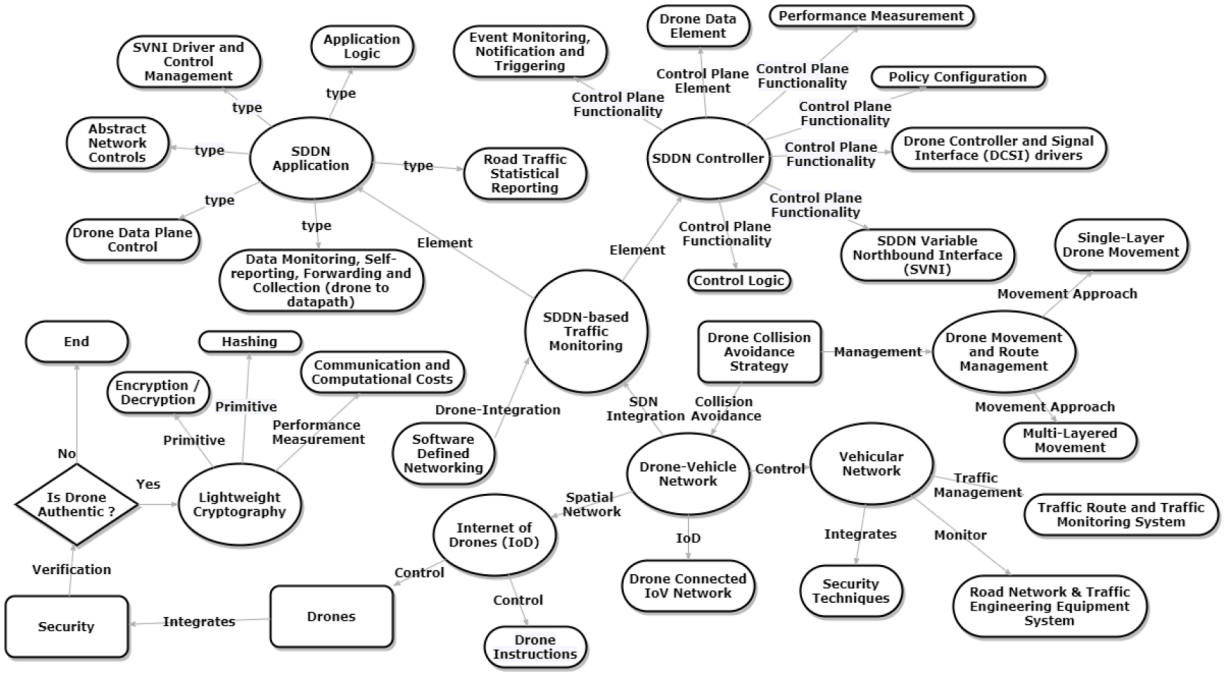


Fig. 1. SDDN-based Traffic Monitoring Ecosystem.

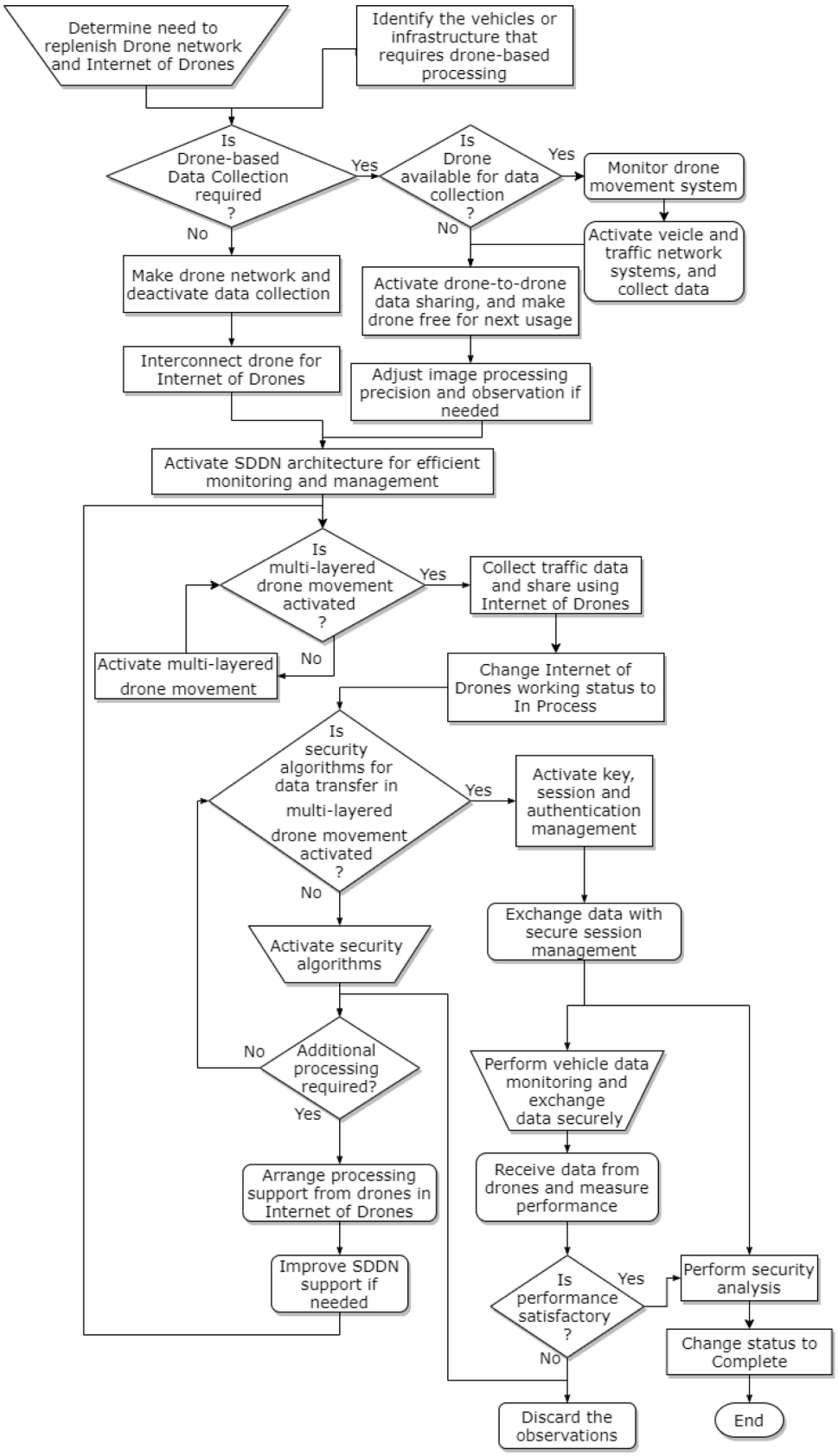
mobile vehicle datasets. These correlations may be found in mobile vehicle datasets. During the process of formulating the technique, these connections were included into the discussion at various points. Evidence for the presence of these correlations between the variables was supplied by the data gathered by the mobile cars as they drove about. Following a thorough examination of the data, it was discovered that these correlations between the different variables do, in fact, exist. The hybrid system that this algorithm was designed for had amassed a significant quantity of data, and the individuals who were responsible for its development started out with the intention of gleaning insight from that store of information. SDN-based frameworks are also proposed in recent times to ensure the security of network. For example, Lahlou et al. [42] proposed an SDN-based framework that is based on SDN, and it is used for successfully recognising and mitigating a broad range of cyber-security threats. Likewise, Polat et al. [43] proposed a scenario to detect and avoid DoS attacks using SDN framework [44,45]. Likewise, various advanced technologies (including quantum computing [46,47], cloud computing [37], artificial intelligence and machine learning [36,37], object detection and collision avoidance, traffic monitoring [48], blockchain [49,50], autonomous systems [51], and many more) can be used in vehicular networks and traffic engineering.

Fig. 1 shows the SDDN-based traffic monitoring ecosystem. The major sub-systems in the SDDN-based traffic monitoring ecosystem include drone-vehicle networks, software-defined networking, SDDN applications and SDDN controllers. The drone-vehicle network is connected with a vehicular network for on-road data collection, transmission to nearby processing units, and generating data analytics. Further, drone-vehicle networks take the help of drone connected IoV networks to operate the drones, avoid collisions, and monitor the traffic. The Internet of Drones (IoD), constituted with multiple drone flying, verified drone authenticity through a lightweight cryptography mechanism to ensure security. SDDN-based traffic monitoring has various applications in vehicular networks. For example, SDDN support collects the abstract view of the vehicular network and ensures control instructions, provides the interface to drive and control the drones for traffic management, and helps in application logic and road traffic statistics and report generation. The major sub-system of SDDB-based traffic monitoring is the SDDN controller. Using SDDN controller, drone data elements can help in monitoring and controlling the drones, network can be monitored, policies can be configured, drone controller and signal interfaces can be driven to direct the drone for specific talk,SVNI interfaces helps in overall management of drone and vehicular networks as per operating instructions.

Critical Analysis: In literature [3,13–18,21,22,25–27,52], drone/UAV-based system for traffic surveillance faces various challenges including (i) heavy computational requirements to analyze the on-road vehicle and traffic conditions, (ii) secure drone-to-drone communication for data exchange and drone network security, (iii) lack of drone-to-drone collision detection mechanisms except LiDAR/Radar/Optical system, (iv) lack of drone-to-drone collision avoidance strategies for small to large scale drone movements and surveillance systems, (v) fixed camera-based or hybrid (both fixed camera and UAV) traffic monitoring are expensive to install, maintain, demand more person hours to finish the job, (vi) a large number of on-road incidents are not reported because of fixed traffic monitoring infrastructure or lack of flexible solution, and (vii) majority of drone-based solutions focus on UAV-based real-time data collection or decision-making rather providing UAV network performance and security-based metrics for data security and overall system performance analysis.

Table 4
Symbols and Notations.

Symbol	Explanation
D_i^j	ith drone moving at jth layer
$L_{D_i^j}^a$	Latitude position of D_i^j
$L_{D_i^j}^o$	Longitude position of D_i^j
T_i	ith traffic engineering equipment
$T_i^{D_i^j}$	ith traffic engineering equipment close to D_i^j
H	Lightweight hash function
$P(.)$	Chebyshev polynomial function
$S_{i,k}$	Session id between two entitites i and k
$K_{(D_i^j,D_k^j,S_{i,k})}^{old}$	Old session key between D_i^j and D_k^j for session $S_{i,k}$
$K_{(D_i^j,D_k^j,S_{i,k})}^{current}$	Current session key between D_i^j and D_k^j for session $S_{i,k}$
C	OpenMP/Cloud center for parallel and distributed computing
$I_{D_i^j}$	Identification of ith drone moving at jth layer
e_i	Random number generated by drone with period $[1,\alpha]$
r_i	Random number generated by drone with period $[1,\beta]$, $\beta \geq \alpha$
σ_i	Random number generated by cloud center with period $\geq \alpha$
$a\|\|b$	Bitwise OR operation is applied between a and b
$\varphi_{D_{i-1}^j}$	Secret of D_{i-1}^j
$(A)^{'}$	Transpose of an entity A
$K_{(C,\ D_{i-1}^j,D_i^j,S_{i-1,i})}^{Public}$	Public key of C generation for $S_{i-1,i}$ session between D_{i-1}^j D_i^j
$K_{(C,\ D_{i-1}^j,D_i^j,S_{i-1,i})}^{Private}$	Private key of C generation for $S_{i-1,i}$ session between D_{i-1}^j D_i^j



(caption on next page)

Fig. 2. Proposed system workflow.

2.1. Symbols and notations

Table 4 shows the symbols and notations used in this paper.

3. Proposed drone based monitoring system

This section illustrates the proposed drone/UAV based distributed traffic monitoring system as shown in Fig. 1. Fig. 2 explains the workflow of the proposed approach in detail.

In the proposed system, the need of a drone-based network and the Internet of Drones are realized to monitor the traffic conditions and infrastructure. If drones are required to collect the data and are available to constitute a multi-layered drone network then vehicles are identified using image processing. Further, vehicle speed, number plate reading and multi-angle identification work to identify the vehicles, and monitor and manage the traffic. The multi-layered drone network is efficiently managed by SDDN-based architecture. Thereafter, security and performance analysis is performed to ensure a better quality of service.

A. drone system

This system consists of a set of drones moving in a particular area. This paper considers drones' movement mapped to roads area. Various sub-systems considered in this system are listed and explained as follows.

1) *Drone network and Internet of Drones*: This sub-system takes care of drones' movement in single or multi-layered strategies. To avoid any collision, single or multi-layered drone movement strategies take care of the drone information system. In the drone information system, each drone maintains distance from its neighboring drones, drone's flying altitude, the number of layers used for traffic monitoring, and data sharing with cloud services, and collision avoidance strategy. In collision avoidance strategy, drones or any object coming in drone's path are initially avoided using LiDAR/Optical/Radar systems. Whereas, layer-based preemptive drone's movement strategies avoid excessive dependency over LiDAR/Optical/Radar system and provide easy integration of other drone-based applications
2) *Federated Data Processing:* In this sub-system, drones are programmed to collect data using sensors or image processing. In a sensor-based data collection system, drones use axle, acoustic, loop detector, doppler, piezoelectric, tube, etc. sensors for on-road traffic data collection. This data is collected from system computers and wireless data transmitters placed alongside the roads. Another way of drone's data collection is through video recording and image processing. On-road vehicle's videos are recorded to identify different traffic flow characteristics such as vehicle speed, length, type, direction, occupancy, estimated weight etc. All of this information is shared with a distributed and parallel computing system for information processing, analytics, visualization, and storage.
3) *Drone-to-drone data sharing*: Single or multiple layered drone movement approaches allow the drones to move either in direction of traffic or the opposite of it. Thus, drone-to-drone data exchange for ad-hoc network construction and increasing the range of information availability would be much easier. This exchange of information should use lightweight mechanisms (security and data sharing) keeping drone's scarcity of resources into consideration.
4) *Image Processing*: In this sub-system, images are extracted from videos for information filtration such as vehicle type, the relative speed with a drone, object, etc. This information is useful for generating the necessary proofs for other associated systems. Fig. 3 shows the image processing system programmed (using Tenflow, OpenCV, CloudXLab and Python) and used in the proposed traffic monitoring system. Fig. 3(a) shows an example of vehicle detection and relative speed of vehicles with drone. Fig. 3(c) shows the single-vehicle detection and inference score (on a scale of 0 to 1). Fig. 3(d) shows image-based multi-object detection when vehicles are parked or stationary. Fig. 3(b) shows that the proposed system is capable of detecting multiple objects (including vehicles) when traffic is moving. Additionally, the inference rate is computed as well. Fig. 3(c) shows the number plate reading system. This is an image-based text reading system that reads all text from the back image of a vehicle. The read text can include advertisements attached to the vehicle, model no. etc. To filter the vehicle registration number, "AA NN AA NNNN" template is followed where 'A' denotes an alphabet and 'N' denotes number from 0 to 9.
5) *SDDN architecture*: The SDDN based distributed architecture using drones is shown in Fig. 4. This architecture is divided into four planes (road-mapped drone network, drone data, controller, and application) and three types of interfaces (DnI, SCDEI, and SCAI). The road-mapped drone's network plane considers drones collision-free movements, and drones mapped to roads for vehicle

(a) On-road vehicle and relative speed detection (example 1)

(b) Stationary multi-object detector and Interference percentage

(c) Number-plate reading system

Fig. 3. Image processing system in the proposed traffic monitoring system (with a maximum vehicle to drone distance of 15 meters).

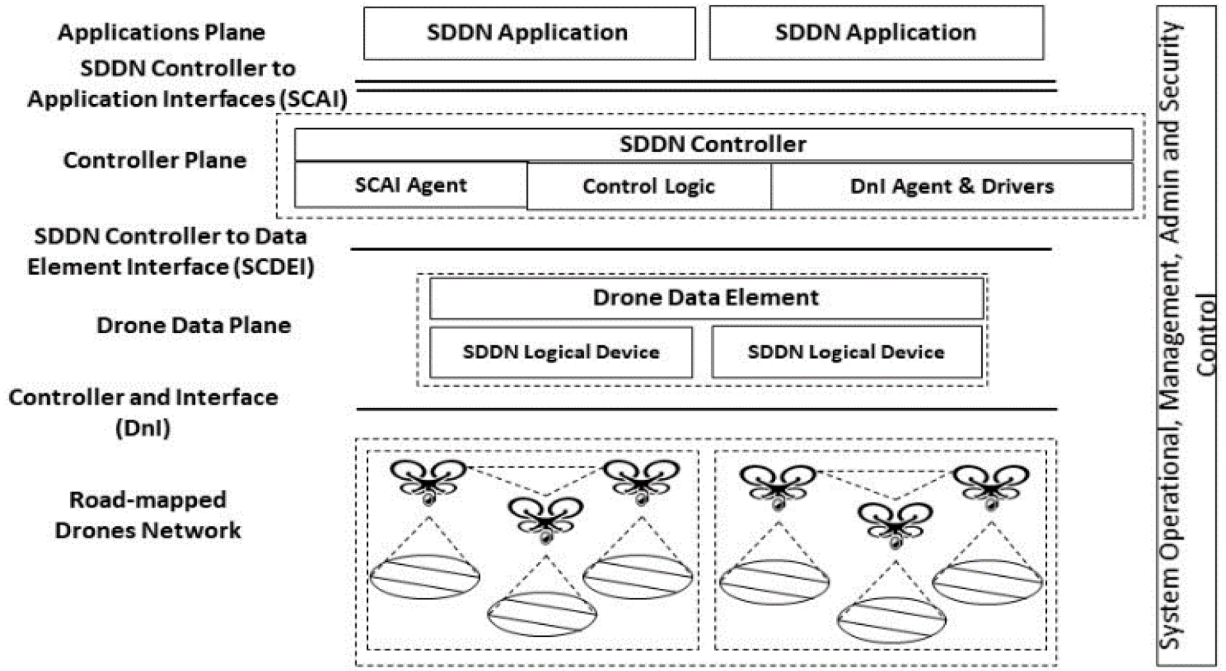


Fig. 4. Proposed SDDN-based architecture for traffic monitoring using drones.

monitoring. The DnI interface collects signal from logical drone data element and instructs the physical drones. This interface helps in collecting and forwarding the required data as well. The drone data plane consists of a drone data element and SDDN logical device (as per OpenFlow 1.3 Specifications). The SDDN logical device operates and program the drones for data, collect the data using DnI interface, and display statistical report. The drone data element is a logical entity The SCDEI interface provides the required information to the SDDN controller for individual drone instruction. The SDDN controller in the controller plane has SCAI agent, control logic, and DnI agent and driver. The SCAI agent compares the collected data with normal operational data and prepares the drone's network and data abstract views. The control logic handles the individual drone's observation and processing capabilities. This entity decides to switch the drone from one layer to another, and accordingly change the functionality. The DnI agent and driver trigger the events observed in the data and pass the information to the SDDN controller for decision making. The SCAI handles the data exchange between the SDDN controller and the user-controllable SDDN application. There are multiple SCAI interfaces for different abstract views of the drone's network. The application plane consists of SDDN applications for users and administrators to have network abstract views and passes the instructions accordingly. Thus, the application plane adds manual control in the SDDN-based drone control system for traffic monitoring.

6) *Data visualization*: This sub-system displays the types and number of unique objects identified. This identification helps in measuring the traffic in a particular area and within a specified time. Further, the trajectory-based vehicle movement helps in predicting future traffic flows and resource planning.

B. Drone-based traffic data collection system

This system consists of the road network, vehicle network, traffic engineering, and drone movement sub-systems. The details of each of these sub-systems are briefly explained as follows.

1) *Road network system*: This system assumes that road-network information is available before deployment and movement of drones for the proposed system.
2) *Vehicle network system*: This system assumes that all on-road vehicles are registered. Additionally, the vehicle type, length, model etc. features are available to identify its category.
3) *Traffic engineering system*: This system is considered to provide road-traffic data through sensor-devices placed inside or over the road. This sensor-based electric wiring system generates electromagnetic waves and forms a loop detector system. This loop detector system analyzes the vehicle, its speed, weight etc. This loop detector system share information with the computer system through an electric meter which is further forwarded to drones through wireless data transmitter.
4) *Drone movement system*: In this sub-system, the drone's movement, collision-free strategy, multi-layering, and image and sensor-based data collection are handled.

C. MPI-distributed and parallel computing

In this paper, MPI is used for distributed and parallel programming. MPI provides standardization, portability, performance opportunities, functionality and availability. MPI interfaces (MPI_Scatter and MPI_Gather) are used to compute hash values for drones. In our proposed system, drones are considered as resource-constraint devices. Thus, MPI-based support systems computes hash values and sends over to drones for implementing security services in Algorithms 4 and 5. Here, MPI interfaces are used in hash computation. In this process, input to process 0 (initial process) is assigned statically and this process uses MPI_Scatter to distribute the input messages among different processes sequentially to compute hash values (using Algorithm 1). Thereafter, MPI_Gather interface is used to collect all hashes (using Algorithm 2). Over multicore or many-core processor architectures, integration of MPI with shared memory, and distributed and parallel programming is possible in OpenMP programming model. This model effectively utilizes the resources and improves computational and internode communication performances.

Fig. 5 shows Hash-computation MPI workflow diagram used in this paper. Initially, n-Hash-computation and m-Hash collection MPI applications are used. Thereafter, the performance of the system is measured using the computing power required to execute applications, the relative computing power of the individual processor, applications execution time (including computation and communication times), application spawning and removal costs, and data redistribution or rearranging costs. The measured system performance is constantly monitored and evaluated to be considered for the proposed system. If the performance is satisfactory (greater than a certain threshold) then more applications are spawned (maximum of n-Hash computation and m-Hash collection) and input to an application are rearranged to generate randomized Hash outputs. Now, if the performance is not satisfactory then applications can be gradually reduced to improve the performance. MPI dynamic (automated) or static (manual) computation processes can be used for application expansion or shrinking.

4. Drone movement algorithms

This section proposes a 3-drone vehicle monitoring system and multi-layer drone's movement and collision avoidance strategies. A 3-drone vehicle monitoring system is designed to identify vehicles from the top, front, and rear-view images. The multi-layered drone's movement and collision avoidance strategy are proposed to have road-mapped drone-movement for data collection and distribution to parallel processing units. The details of both sub-systems are explained as follows.

A. 3-drone vehicle monitoring system

This section proposes a 3-drone vehicle monitoring system. Here, three drones (at a distance of d_1 and d_2) observing a single vehicle at a time in the pre-defined area move their camera at an angle θ_1, θ_2 and θ_3 to capture front, top, and read view images as shown in Fig. 6. Further, angles $\varnothing_1$ and $\varnothing_2$ are considered to read the number-plates and collect the registration number data. The collected information is analyzed for measuring the inference rate and results show that the image-based vehicle identification system has an inference rate greater than 88% if the image is collected at a maximum distance of 10 m. Fig. 7 shows different views of 3-Drone Vehicle Monitoring System. The side-view of the proposed UAV based vehicle monitoring system is illustrated in Fig. 7(a). In the multi-level system, multiple-drones handle the data and shred the data processing load. Layer-1 drones collect on-road vehicle information and pass it to layer-2 drones. Layer-2 drones collect the information from layer-1 drones only and compile the data for layer-3 drones. Layer-3 drones are connected with SDDN and private cloud systems for parallel and distributed processing, and data handling. The front view of the proposed system is depicted in Fig. 7(b). It shows that three drones are handling each vehicle one-by-one in each road-lane at layer 1. Thereafter, data handling is distributed among drones at layer-2 and layer-3 before openMP-based data handling.

B. Multi-layer drone system for traffic monitoring

This sub-section explains the multi-layer drone system used for traffic monitoring. Here, collision avoidance strategy for drones in multi-layered environment is proposed. Multi-layered drone-movement and collision avoidance approaches are explained using Algorithm 3 in detail. In this algorithm, LiDAR/Optical detector and distance based collision avoidance strategy is proposed.

Algorithm 1
HashComputationWithScatter(Message[], InitialVector, recv_buffer).

```
1. Begin
2. s ← Size() and r ← Rank()
3. repeat(1)
4. recv_buffer=MPI.linspace(s,r)
5. hash ← MPI_Scatter(Message, recv_buffer,root=0)
6. hash_array ← hash_array.Append()
7. return hash_array
8. End
```

Algorithm 2
HashCollectionWithGather(Message[], InitialVector, send_buffer, recv_buffer).

1. **Begin**
2. s ← Size() and r ← Rank()
3. repeat(1)
4. send_buffer=MPI.linspace(s,r)
5. hash ← MPI_Gather(send_buffer, recv_buffer, root=0)
6. return hash
7. **End**

Algorithm 3
Proposed multi-layered drone-based collision-avoidance and 3-drone vehicle monitoring strategy.

Goal: To create 3-drone internet and collision-free drones' movement strategy for traffic monitoring.
1. **Begin**
2. Construct a group of 3-drones $\{(D_1^1, D_2^1, D_3^1), (D_4^1, D_5^1, D_6^1), \ldots, (D_{n-2}^1, D_{n-1}^1, D_n^1)\}$ and deploy parallel to the road at minimum distance of D between any two groups of drones
3. **While** (Any group out of $\{(D_1^1, D_2^1, D_3^1), (D_4^1, D_5^1, D_6^1), \ldots, (D_{n-2}^1, D_{n-1}^1, D_n^1)\}$is flying) **do:**
4. **For each** $D_n^1 \in (D_{n-2}^1, D_{n-1}^1, D_n^1)$:
5. D_n^1 uses LiDAR/Optical detector to maintain a minimum distance of D with every other object
6. D_{n-2}^1 maintain a minimum distance of d_1 with D_{n-1}^1, and D_{n-1}^1 maintain a minimum distance of d_2 with D_n^1
7. **If** any distance is less than D **then**
8. Move colliding drone apart and maintain a minimum distance of D
9. **If** drone movement is not possible then
10. Land $(D_{n-2}^1, D_{n-1}^1, D_n^1)$
11. **Else**
12. Collect Sensor or image-based data
13. **End If**
14. **End For**
15. **End While**
16. **End**

5. Drone-to-drone data transfer and security algorithms

This section proposes two drone-to-drone session key generation algorithms (algorithm 4 and algorithm 5) that can be used for securely transmitting the data while maintaining its confidentiality. Algorithm 4 proposes multi-round challenge-response verification approach for drone-to-drone session key generation and renewal for data encryption and decryption. In this algorithm, temporary challenges are generated in step 1 to step 9. From step 10 to step 14, challenge sending, verification, and renewal is processed until drone is considered authentic. Algorithm 5 uses elliptic curve cryptography for same purpose. In algorithm 5, temporary challenges are generated using elliptic curve cryptography in step 1 to step 8. From step 9 to step 16, challenge sending, verification, and renewal is processed (using elliptic curve cryptography) until drone is considered authentic.

The comparative cost analysis of proposed algorithms in key generation and renewal is shown in Table 4. Since algorithms use more than one type of logic gates for different operations, the hardware cost (also measure circuit size) is measured in NAND gate equivalent (GE) to reflect the requirements of the actual area. Assuming, two-input NAND, NOR, AND, OR and XOR gates require 1, 1, 1.25, 1.25, and 2.25 GEs, respectively and three-input NAND, NOR, AND, OR, XOR, MAJ and MUX gates require 1.25, 1.5, 1.5, 1.75, 4.0, 2.25 and 2.50 GEs respectively, Table 5 shows the total computational and communication cost in terms of GEs. Here, single multiplication requires 6 AND and 2 XOR gates. Additionally, a typical elliptic curve implementation requires 4548 (arithmetic unit), 11205 (memory) and 2368 (control logic) GEs for 163-bit scalar key size. The comparative analysis of the key generation and renewal process (as shown in Table 4) shows that our proposed algorithms require fewer GEs than state-of-the-art protocols. Thus, our proposed algorithms are lightweight in nature. Algorithm 5 requires lesser GEs (total cost) compared to algorithm 4. Although the computational cost of algorithm 5 is slightly higher than algorithm 4, the communicational cost is much lower. Overall, algorithm 5 shows better performance in terms of GEs compared to algorithm 4 or state-of-the-art approaches. In case computational cost is the only criteria for performance evaluation, then algorithm 4 is preferred over algorithm 5. For example, if the size of the data packet is large or heavy cryptographic primitive is used for security purposes, the computational cost will be high. In such scenarios, algorithm 4 performs slightly better compared to algorithm 5.

6. Security analysis

In this section, our proposed drone-to-drone session key generation and renewal algorithms are analyzed against various attacks. More details of security analysis are as follows.

Algorithm 4
Proposed arithmetic operations-based drone-to-drone session key generation and renewal for authentication.

Goal: To generate session key for secure data transfer in finite field with airthmetic operations.

1. **Begin**
2. $D_{i-1}^{j} \rightarrow D_{i}^{j}$: r_1
3. D_{i}^{j} computes:
a. $\text{temp}_1=\text{H}(I_{D_{i-1}^{j}}) \oplus \text{e}_1 \oplus \text{r}_1$, $\text{temp}_2=P_{r_1,e_1}(K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})})$, $\text{temp}_3=K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})} \oplus \text{e}_1$
4. $D_{i}^{j} \rightarrow D_{i-1}^{j}$: temp_1, temp_2, temp_3
5. $D_{i}^{j} \rightarrow D_{i+1}^{j}$: r_2
6. D_{i+1}^{j} computes:
a. $temp_1'=\text{H}(I_{D_i^j}) \oplus \text{e}_2 \oplus \text{r}_2$, $temp_2'=P_{r_2,e_2}(K^{current}_{(D_i^j,D_{i+1}^j,S_{i,i+1})})$, $temp_3'=K^{current}_{(D_i^j,D_{i+1}^j,S_{i,i+1})} \oplus \text{e}_2$
7. $D_{i+1}^{j} \rightarrow D_{i}^{j}$: $temp_1'$, $temp_2'$, $temp_3'$
8. $D_{i-1}^{j} \rightarrow C$: r_1, temp_1, temp_2, temp_3
9. $D_{i}^{j} \rightarrow C$: r_2 , $temp_1'$, $temp_2'$, $temp_3'$
10. C computes:
a. $K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})} = \text{temp}_1 \oplus \text{temp}_3 \oplus \text{r}_1$, $\text{temp}_4=\text{H}(I_{D_{i-1}^{j}}) \oplus K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})}$
b. **if** temp_4 record exist **then**
c. fetch $\text{H}(I_{D_{i-1}^{j}})$, $K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})}$, $K^{old}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})}$
d. $\text{temp}_5=\text{temp}_1 \oplus \text{H}(I_{D_{i-1}^{j}}) \oplus \text{r}_1$, $\text{temp}_6=\text{H}(I_{D_{i-1}^{j}}) \oplus \text{r}_1 \oplus \sigma_1$
e. **if** temp_2 equals to $P_{r_1}(P_{e_1}(K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})}))$ **then**
f. $\text{temp}_7 = P_{dc_1,e_1}(K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})})$, $K^{old}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})}=K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})}$
g. $K^{Current}_{Session}=K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})} \oplus (\text{e}_1 \| \sigma_1)$
h. **else if** temp_2 equals to $P_{r_1}(P_{e_1}(K^{old}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})}))$ **then**
i. $\text{temp}_7 = P_{dc_1,e_1}(K^{old}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})})$, $K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})}=K^{old}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})} \oplus (\sigma_1 \| \text{e}_1)$
j. **else**
k. communication is unauthentic
l. **End if**
11. $C \rightarrow D_{i}^{j}$: $temp_6$, $temp_7$
12. $D_{i}^{j} \rightarrow D_{i-1}^{j}$: $temp_6$, $temp_7$
13. D_{i+1}^{j} computes:
a. $\text{dc}_1=temp_6 \oplus \text{H(ID)} \oplus \text{r}_1$
b. **if** $temp_7$ equals to $P_{dc_1,e_1}(K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})})$ then
c. $K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})} = K^{current}_{(D_{i-1}^{j},D_{i}^{j},S_{i-1,i})} \oplus (e_1 \| \sigma_1)$
d. **End if**
14. C computes:
a. $K^{current}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})}=temp_1' \oplus temp_{1=3}' \oplus r_2$
b. $temp_4'=\text{H}(I_{D_i^j}) \oplus K^{current}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})}$
c. **if** $temp_4'$ record exist **then**
d. fetch $\text{H}(I_{D_i^j})$, $K^{current}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})}$, $K^{old}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})}$
e. $temp_5' = temp_1' \oplus \text{H}(I_{D_i^j}) \oplus r_2$, $temp_6' = \text{H}(I_{D_i^j}) \oplus r_2 \oplus \sigma_2$
f. **if** $temp_2'$ equals to $P_{r_2}(P_{e_2}(K^{current}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})}))$ **then**
g. $temp_7' = P_{dc_1,e_1}(K^{current}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})})$
h. $K^{old}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})}=K^{current}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})}$
i. $K^{Current}_{Session}=K^{current}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})} \oplus (\text{e}_2 \| \sigma_2)$
j. **else if** $temp_2'$ equals to $P_{r_2}(P_{e_2}(K^{old}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})}))$ **then**
k. $\text{temp}_7 = P_{\sigma_2,e_2}(K^{old}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})})$
l. $K^{current}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})}=K^{old}_{(D_{i}^{j},D_{i+1}^{j},S_{i,i+1})} \oplus (\sigma_2 \| \text{e}_2)$
m. **else**
n. communication is unauthentic
o. **End if**
15. **End**

Algorithm 5
Proposed ellipctic curve cryptosystem-based drone-to-drone session key generation and renewal for authentication.

Goal: To generate session key for secure data transfer in finite field with elliptic curve cryptosystem.

1. **Begin**
2. $D_{i-1}^{j} \rightarrow D_{i}^{j}$: e_1P, e_2P
3. $D_{i}^{j} \rightarrow D_{i+1}^{j}$: e_3P, e_4P
4. $D_{i}^{j} \rightarrow D_{i-1}^{j}$: r_1
5. $D_{i+1}^{j} \rightarrow D_{i}^{j}$: r_2
6. $D_{i-1}^{j} \rightarrow C$: e_1P, e_2P, r_1
7. $D_{i}^{j} \rightarrow C$: e_3P, e_4P, r_2
8. C computes:
a. $\text{temp}_1 = (e_1 + r_1 I_{D_{i-1}^{j}}{}') \, K^{Private}_{(C, D_{i-1}^{j}, D_{i}^{j}, S_{i-1,i})}$, $\text{temp}_2 = (e_2 I_{D_{i-1}^{j}}{}' + r_1 \varphi_{D_{i-1}^{j}}{}') \, K^{Private}_{(C, D_{i-1}^{j}, D_{i}^{j}, S_{i-1,i})}$,
$\text{temp}_3 = (e_3 + r_2 I_{D_{i}^{j}}{}') \, K^{Private}_{(C, D_{i}^{j}, D_{i+1}^{j}, S_{i,i+1})}$, $\text{temp}_4 = (e_3 I_{D_{i}^{j}}{}' + r_2 \varphi_{D_{i}^{j}}{}') \, K^{Private}_{(C, D_{i}^{j}, D_{i+1}^{j}, S_{i,i+1})}$
9. $C \rightarrow D_{i-1}^{j}$: $\text{temp}_1, \text{temp}_2$
10. $C \rightarrow D_{i}^{j}$: temp_3, temp4
11. D_{i-1}^{j} computes:
12. **If** $I_{D_{i-1}^{j}} == ((K^{Public}_{(C, D_{i-1}^{j}, D_{i}^{j}, S_{i-1,i})})^{-1} \, \text{temp}_1 - e_1P) r_1{}^{-1}$ **then**
a. $K^{Current}_{(C, D_{i-1}^{j}, D_{i}^{j}, S_{i-1,i})} = ((K^{Public}_{(C, D_{i-1}^{j}, D_{i}^{j}, S_{i-1,i})})^{-1} \, \text{temp}_2 - I_{D_{i-1}^{j}}{}' e_2P) \, r_1{}^{-1}$
b. **End If**
13. $D_{i-1}^{j} \rightarrow D_{i}^{j}$: $K^{Current}_{(C, D_{i-1}^{j}, D_{i}^{j}, S_{i-1,i})}$
14. D_{i}^{j} computes:
15. **If** $I_{D_{i}^{j}} == ((K^{Public}_{(C, D_{i}^{j}, D_{i+1}^{j}, S_{i,i+1})})^{-1} \, \text{temp}_3 - e_3P) r_2{}^{-1}$ **then**
a. $K^{Current}_{(C, D_{i}^{j}, D_{i+1}^{j}, S_{i,i+1})} = ((K^{Public}_{(C, D_{i}^{j}, D_{i+1}^{j}, S_{i,i+1})})^{-1} \, \text{temp}_4 - I_{D_{i}^{j}}{}' e_4P) \, r_2{}^{-1}$
b. **End If**
16. $D_{i}^{j} \rightarrow D_{i+1}^{j}$: $K^{Current}_{(C, D_{i}^{j}, D_{i+1}^{j}, S_{i,i+1})}$
17. **End**

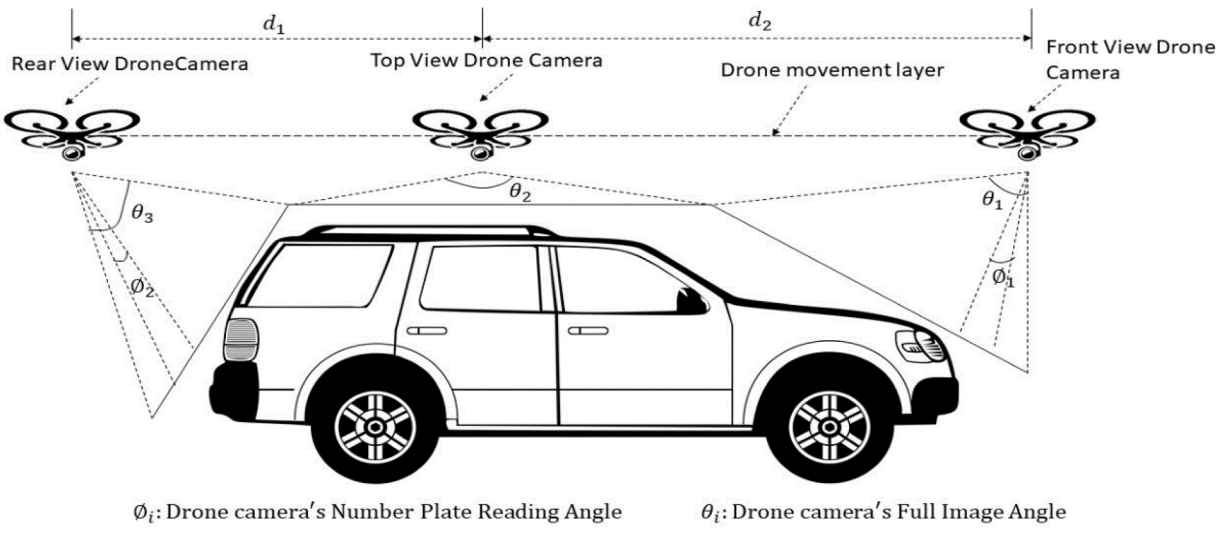


Fig. 5. Hash computation MPI Workflow.

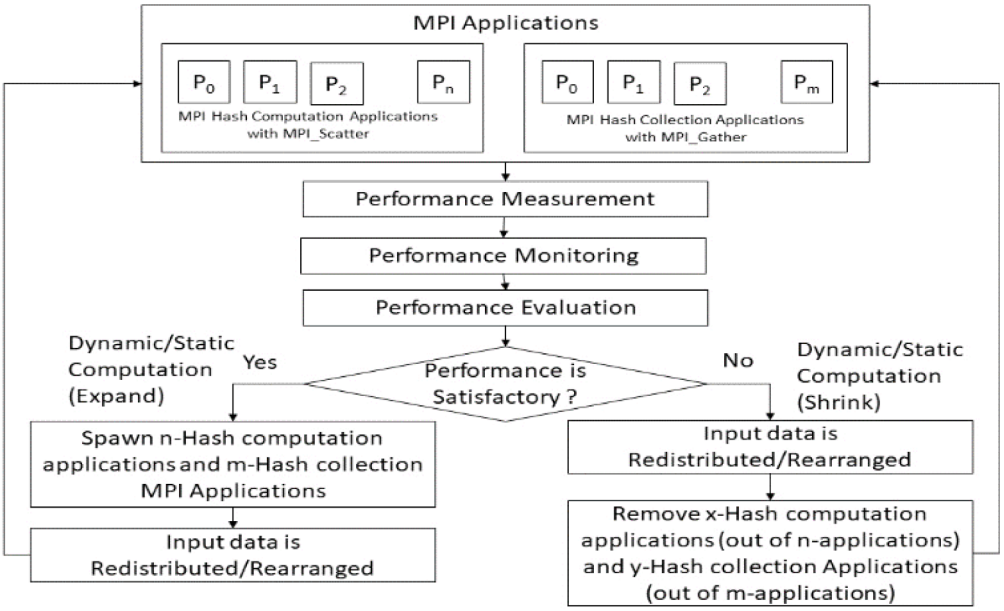


Fig. 6. 3-drone vehicle monitoring system.

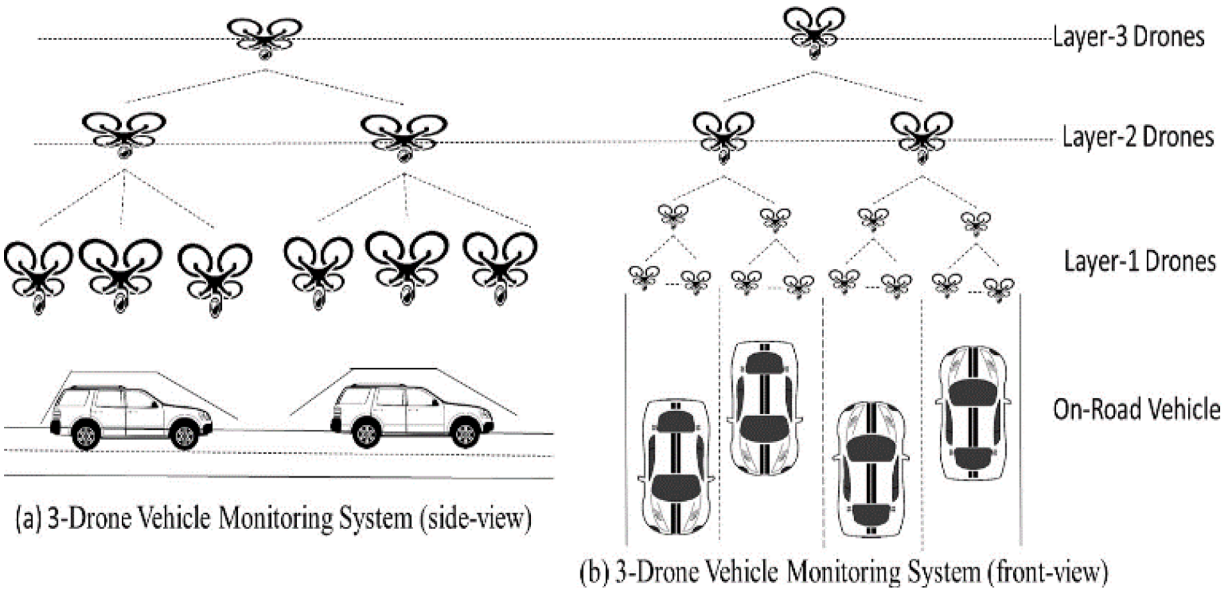


Fig. 7. 3-Drone Vehicle Monitoring System (side-view and front view).

Table 5
Comparative cost analysis of key generation and renewal process*.

Algorithm	Computational Cost	Communicational Cost	Total Cost (GEs)
Algorithm 4	8M=136 GEs	27X+11h+15p=108+11000+255= 11,363 GEs	29,620 GEs
Algorithm 5	10M=170 GEs	34M+4D=57+88 = 145GEs	18,436 GEs
Singh et al. [53]	6M + 7X +1 h = 102+28+1000 = 1130 GEs	31X+9h+7E=124+9000+3500 = 12,624 GEs	31,875 GEs
Zhang et al. [54]	1h+1X=1000+4 = 1004GEs	69O+14X+22h=120.75+56+22000 ≈ 22,177GEs	42,302 GEs

* M=Multiplication, D=Doubling, X=XOR, h=hash computation cost, p= Chebyshev polynomial function computation cost, O=OR, E=exponentiation (with modulus)

Strong Secrecy: It means that the protocol is secure against attack in case of change in secret values. In algorithm 4 and algorithm 5, private and public keys, and temporary values change frequently. Thus, the proposed protocols will be considered secure if old and new secret values should not affect the working behavior of the protocols. In this analysis, ProVerif toolkit is used to vary the attacker's ability to learn secret information partially. The conditions when the security proofs are considered in the sense of strong secrecy are:

- Attacker has learned some components of a pair value, for instance, but he is not able to fetch the whole pair values because he does not have other components of the pair. For example, ($temp_1$, $temp_2$, $temp_3$) is a pair of values that are exchanged between drones in algorithm 4. Now, if the attacker's ability to learn any information about the pair is high, provided one component of the pair (e.g. $temp_1$) is known to him, then it is not considered in the sense to have strong secrecy. Now, the concept of strong secrecy is important when components in the pair consist of known values rather than unknown values. Let the real number system $(R; +; -; .; \oplus; ||; E_{MI}; E_{PD}; E_{PA})$ is defined with addition ($+$), substraction (-), multiplication (.), XOR ($\oplus$), OR ($||$), multiplicative inverse (E_{MI}), elliptic point doubling (E_{PD}) and elliptic point addition (E_{PA}) operations in finite field (Z_n) with size n. Consider, for instance, a process that uses ($temp_1$, $temp_2$, $temp_3$) pair in Z_3. Now, each component in this pair can have a value 0, 1 or 2. Thus, it can be assumed that $temp_i \in \{temp1,\ temp2,\ temp3\}$ is strongly secret with the probabilities (Eq. (1)):

$$P\left\{\frac{0}{temp_i}\right\} \approx P\left\{\frac{1}{temp_i}\right\} \approx P\left\{\frac{2}{temp_i}\right\} \tag{1}$$

i.e. attacker is not able to determine whether $temp_i$ is 0, 1 or 2. Here, $\frac{n}{temp_i}$ represent that $temp_i$ is assigned a value n. This assumption may be wrong because n is very small. A large value of n can increase the $temp_i$ pair values which in-turn increases the confidence in strong secrecy. Now, consider an instance when an attacker is continuously trying with multiple attempts. In this case, strong secret probabilities can be written as (Eqs. (2) and (3)):

$$P\left\{\frac{0}{temp_i}\right\} + P\left\{\frac{1}{temp_i}\right\} + P\left\{\frac{2}{temp_i}\right\} + P\left\{\frac{3}{temp_i}\right\} + P\left\{\frac{4}{temp_i}\right\} + \ldots\left\{\frac{c_i}{temp_i}\right\} + \ldots \infty \tag{2}$$

$$\sum_{i=1}^{\infty} P\left\{\frac{c_i}{temp_i}\right\} = 1 \tag{3}$$

Here, c_i is the i^{th} component in the pair and defines unique probability distribution by association (A) as (Eq. (4)):

$$P(A) = \sum_{\frac{c_i}{temp_i} \in A} P\left\{\frac{c_i}{temp_i}\right\} \tag{4}$$

For an attacker, $P(A)$ is greater than 0 but very small as there is randomness in component guessing. Thus, it would be difficult to perform an attack. With a large number of guesses, establishing an association among continuous attempts would be difficult. Thus, the chances of an attack are negligible.

- An attacker has learnt some component of a pair value, but he cannot distinguish changes in the values of $temp_i$. Under this condition, a process is considered to have strong secrecy if (Eq. (5)):

$$P\left\{\frac{temp_1'}{temp_1}, \frac{temp_2'}{temp_2}, \ldots, \frac{temp_n'}{temp_n}\right\} \approx P\left\{\frac{temp_1''}{temp_1}, \frac{temp'^{\prime}_2}{temp_2}, \ldots, \frac{temp_n''}{temp_n}\right\} \tag{5}$$

for all component values $temp_1'$, $temp_2'$,..., $temp_n'$, $temp_1$,$temp_2$, ..., $temp_n$,$temp_1''$, $temp_2''$,..., $temp_n''$. This means, the attacker can change $temp_1$,$temp_2$, ..., $temp_n$ to any old or new values $temp_1'$, $temp_2'$, ..., $temp_n'$, $temp_1''$, $temp_2''$,..., $temp_n''$} but he is unable to get pair. In case, attacker make multiple attempts and try to build association among attempts then probability distribution by association becomes (Eq. (6)):

$$P(A) = P\left\{\frac{temp'_1}{temp_1}, \frac{temp'_2}{temp_2}, \ldots, \frac{temp'_n}{temp_n}\right\} + P\left\{\frac{temp''_1}{temp_1}, \frac{temp''_2}{temp_2}, \ldots, \frac{temp''_n}{temp_n}\right\} + P\left\{\frac{temp'''_1}{temp_1}, \frac{temp'''_2}{temp_2}, \ldots, \frac{temp'''_n}{temp_n}\right\} + \ldots + \ldots\infty \tag{6}$$

For an attacker, $P(A)$ is greater than 0 but very small as there is randomness in component guessing. Thus, it would be difficult to perform an attack. With a large number of guesses, establishing an association among continuous attempts would be difficult. Thus, the chances of an attack are negligible.

- Attacker has learnt some component of a pair value and he has the ability to send multiple queries for one component in its single execution but he cannot distinguish changes in the values of $temp_i$. Under this condition, a process is considered to have strong secrecy with m-attacker queries per component if (Eq. (7)):

$$P\left\{\frac{temp'_{1,1}}{temp_{1,1}}, \frac{temp'_{2,1}}{temp_{2,1}}, \ldots, \frac{temp'_{2,m}}{temp_{2,m}}, \ldots, \frac{temp'_{n,m}}{temp_{n,m}}\right\} \approx P\left\{\frac{temp''_{1,1}}{temp_{1,1}}, \frac{temp'^{'}_{2,1}}{temp_{2,1}}, \ldots, \frac{temp''_{2,m}}{temp_{2,m}}, \ldots, \frac{temp''_{n,m}}{temp_{n,m}}\right\} \tag{7}$$

This means, attacker can change any component with multiple queries $temp_{1,1}, temp_{1,2}, \ldots, temp_{1,m}, \ldots, temp_{n,m}$ to $temp'_{1,1}$, $temp'_{1,2}$, …, $temp'_{1,m}$, … $temp'_{n.m}$ or $temp'_{1,1}$, $temp'_{1,2}$, …, $temp'_{1,m}$, … $temp'_{n.m}$ but he is unable to get pair.

Random Number Matching/Guess: In both algorithms, random numbers are selected within finite field. Assume that anytime an adversary picks up some random number that may or may not be different from currently used number in algorithm execution. Now, the probability (Eq. (8)) that an adversary gets the r-random number of possible matches with honest drone's random number in y-attempts is p.

$$p = 1 - \left(\frac{r}{r} \text{x} \frac{r-1}{r} \text{x} \frac{r-2}{r} \text{x} \ \ldots \ \text{x} \frac{r-(y-1)}{r}\right) \tag{8}$$

Thus, this probability follows (Eq. (9)):

$$0 \le p \le 1 - \left(1 - \frac{y-1}{r}\right)^{y} \tag{9}$$

For large value of y, Eq. (5) can be re-written as (Eq. (10)):

$$0 \le p \le \frac{y(y-1)}{r} \tag{10}$$

Since the adversary's attempt, 'y' is a polynomial, and if the random number 'r' is exponential, then p is negligible in the finite field. In conclusion, an adversary will not have the advantage of randomly selecting a number and matching it with a random number in use. In algorithm 2, two consecutive steps select random numbers (step 3 and step 4). Assuming that there are n-random numbers and an adversary selects 2r random numbers at random (2r<n), Then the probability that an adversary did not find a match after trying $\binom{2n}{2r}$ ways to choose 2r random numbers from 2n. Let the ways to choose matching random numbers be represented as:

$$\left(r_1^1 r_2^1\right)\left(r_1^2 r_2^2\right)\ldots\left(r_1^n r_2^n\right)$$

No matching occurs if the adversary selects only one random number from each 'row' of consecutive random numbers. Here, there are $\binom{n}{2r}$ ways to choose one random number from each row. In this selection, two possible options are available with adversaries (r_1^* or r_2^*). In a result, 2^{2r} ways are available to select step 3 and step 4 random numbers. Thus, the probability that an adversary does not find the selected random number matched with the random number in use (in step 3 and step 4) becomes (Eq. (11)):

$$P(not\ matching) = \frac{\binom{n}{2r}2^{2r}}{\binom{2n}{2r}} \tag{11}$$

Eq. (7) shows that it would be difficult for an adversary to select two random numbers in two consecutive steps and find a match with the algorithm in execution. For example, if an adversary challenge is for (5, 2) then the probability of not matching is 0.38 (or matching is 0.62), i.e. chances of attack is higher. Now, if the adversary challenge is (1000, 2), then the chances of an attack are almost zero. In conclusion, if a large number of bits are used for challenge or primitive generation or verification, then the chances of an attack are negligible.

Dictionary, brute force and guessing attacks: Algorithms may use weak secrets, that is, values lie within a short finite field (e.g. Z_3 is a small field). Algorithms with weak secrets can easily be subject to a dictionary, brute force or guessing attacks, whereby an attacker passively observes the transactions or actively participate in algorithm execution and then has the ability to elongate the execution time at either side (source or destination) or any intermediate temporary computing side. This elongation provides time to the attacker in enumerating a dictionary of component values, verify each component, identify the correct one and use it in further computations. In ProVerif toolkit, the algorithm's protection from dictionary, brute force and guessing attacks can be tested (using weak secret $temp_i$). With this syntax, ProVerif can test whether attacker can distinguish between a successful or unsuccessful attempt or not. It has been observed that for the proposed algorithms, the attacker is unable to distinguish this.

False attacks with processes observational equivalences: In both algorithms (algorithm 1 and algorithm 2), there are multiple 'if' conditions that have the almost same structure but differ in the choice of variables. In process observational equivalences, an attacker may choose one structure while a normal process can choose another. This can lead to "false attacks" in which an attacker may have applied a wrong execution but he can get the observational equivalences. For example, the process observational equivalences for two steps in different 'if' conditions can be represented as (Eq. (12)):

$$\begin{aligned}
&\mathbf{P}\left(K^{current}_{(D^j_{i-1},D^j_i,S_{i-1,i})},\mathbf{e1}\|\ \sigma_1,\oplus\right)\mathbf{orP}\left(K^{old}_{(D^j_{i-1},D^j_i,S_{i-1,i})},\sigma_1\|\ \mathbf{e1},\oplus\right)\\
&\approx \mathbf{P}\left(K^{old}_{(D^j_{i-1},D^j_i,S_{i-1,i})},\mathbf{e1}\|\ \sigma_1,\oplus\right)\mathbf{orP}\left(K^{current}_{(D^j_{i-1},D^j_i,S_{i-1,i})},\sigma_1\|\ \mathbf{e1},\oplus\right)\\
&\approx \mathbf{P}\left(K^{old}_{(D^j_{i-1},D^j_i,S_{i-1,i})},\mathbf{e1}\|\ \sigma_1,\oplus\right)\mathbf{orP}\left(K^{new}_{(D^j_{i-1},D^j_i,S_{i-1,i})},\sigma_1\|\ \mathbf{e1},\oplus\right)\\
&\approx \mathbf{P}\left(K^{new}_{(D^j_{i-1},D^j_i,S_{i-1,i})},\mathbf{e1}\|\ \sigma_1,\oplus\right)\mathbf{orP}\left(K^{old}_{(D^j_{i-1},D^j_i,S_{i-1,i})},\sigma_1\|\ \mathbf{e1},\oplus\right)
\end{aligned} \tag{12}$$

With observational equivalences, the chances of false attack increases. ProVerif execution shows that the proposed algorithms are protected from "false attacks" because of selecting large field size and lesser chances of weak secrets.

- **Real and random authenticated key matching attack**: This attack occurs when the exchanged key in the algorithm is indistinguishable from the attacker's random key selection. ProVerif formal verification process applies several test queries with different conditions that either return the real key with honest participants or it return the real or random key in presence of dishonest participants.
- **Other security aspects**: Each phase of algorithm 4 and algorithm 5 connections has their own correctness and security objectives. For example, during initial phases of communication, both algorithms choose a random number that is consistent with the selected finite field, the key exchange produces a secret session key; keys are updated regularly and so on. These intermediate security goals are important for both algorithms. However, the traffic monitoring application and data exchange mechanism is not much associated with the internal security structure. Consequently, the security goals are for messages to exchange between honest and authenticated drones, that is, for those drone devices whose secret session keys are unknown to the attacker. Assuming that drones are authenticated devices, but they do not know whether they are talking to other honest device or an attacker, the security goals stated in the algorithm's security model are:
- **Data secrecy**: If drone-based traffic monitoring application's data sent over a session between honest drone devices and/or with MPI-based computing centre, then the confidentiality of this message is maintained from an attacker using elliptic curve cryptography operations. This confidentiality ensures that the attacker cannot break the cryptography structure used in the key exchange mechanism.
- **Forward and backward secrecy**: In proposed algorithms, strong secrecy is maintained even the private, public or session keys are given to the adversary after the current session expire and the session keys ($K^{Current}_{(C,\ D^j_{i-1},D^j_i,S_{i-1,i})}$) are deleted from every type of device available in the network.
- **Message authentication**: It states that if drone has exchanged the traffic monitoring data or data associated to generate session keys then it must have sent this data in consecutive sessions (with similar parameters like H, $K^{Current}_{(C,\ D^j_{i-1},D^j_i,S_{i-1,i})}$, $K^{old}_{(D^j_{i-1},D^j_i,S_{i-1,i})}$, $K^{Public}_{(C,\ \ D^j_i,D^j_{i+1},S_{i,i+1})}$).
- **Replay attack**: To avoid replay attack, it should be ensured that any application data sent in one session (or current session) can be accepted at most once by drone devices. Data sent over TCP can maintain the record to handle data loss or drop packets.
- **Weak hash function**: Both algorithms use hash functions for key derivations. The use of lightweight hash functions that requires GEs under 1000 GE with strong collision resistant properties can avoid birthday paradox or other exploitable attacks.
- **Correctness**: If drones in the IoDn complete an exchange (using any algorithm) then same secret session keys are derived at both sides in one session. Whereas, distinct keys are generated in distinct sessions.
- **Mutual distinctiveness**: It is ensured in a proposed algorithm that if one drone shares one session key with another drone, then no other drone should be able to learn anything about key from both sides.

```
460 rules inserted. The rule base cobtains 391 rules. 17 rules in the queue.
Starting query not attacker(svlueDi[])
RESULT not attacker(Divalue[]) is true.
Starting query not attacker(svlueDj[])
RESULT not attacker(Djvalue[]) is true.
----------------------------------------------------------------
Verification summary:
Query ini-event(Dend(Di))==>ini-event(Dbegin(Dj)) is true.
Query ini-event(Dend(Dj))==>ini-event(Dbegin(Di)) is true.
Query not attacker(svalueDi[]) is true.
Query not attacker(svalueDj[]) is true.
Query not attacker(svalueK[]) is true.
```

Fig. 8. Outcomes of ProVerif formal model for algorithm-4.

- **Identity hiding**: In both proposed algorithms, active or passive adversary will not be able to learn about the drone identity ($I_{D_i^j}$) in any exchange or session. This is made possible by not exchanging the identities openly rather in encrypted form only. The process (P_i) that executes the algorithm preserves the secrecy of drones if and only if the role of authentic drone ($R_{authentic}^{D_i^j}$) is disjoint from the role of attacker ($R_{attacker}^{D_i^j}$) such that $f_n(D_i^j) \cup f_n(P_i) \subseteq R_{authentic}^{D_i^j} \cup R_{attacker}^{D_i^j}$ Here, $f_n(D_i^j)$ and $f_n(P_i)$ represents the functions (any) performed by D_i^j and P_i respectively. In any trace $T_r = (R_{authentic}^{D_i^j}, R_{attacker}^{D_i^j})$, T_r will not disclose D_i^j. This has been formally verified in the next section using ProVerif toolkit.

7. Security analysis

This section presents the security and performance results measured using ProVerif [55], AnyLogic [56] and JaamSim simulators [57]. More details of these results and analysis are presented as follows.

A. Formal security verification

Fig. 8 shows the algorithm 4's formal model verification using ProVerif. Similar results are observed for algorithm 5. In ProVerif, description of the algorithm is provided to verify the communications and various security properties using pi calculus. ProVerif translates the algorithms to Horn clauses and analysis, whether the desired security properties hold or not. The major observations in this verification process are discussed as follows:

- Algorithm 4 and algorithm 5 hides the drone identities during any form of their trace. Thus, both algorithms preserve the secrecy of drones.
- Algorithms are protected from chosen cipher text attack, i.e. if an attacker is having cipher text and the key and it tries to obtain the plaintext, then it is discarded during the algorithm execution because the attacker additionally needs to know some or all of the temporary variable values.
- Algorithms are protected from message replay attack because drones in communication authenticate each other and message cannot be changed during communication because of the use of one way hash function.
- As it is difficult to trace the role of any authentic drone, it is not possible to find any false authentication correspondence or secrecy disclosure. Thus, both algorithms are safe and chances of attacks are probabilistically negligible.
- Results show that the key exchange is protected from man-in-the-middle attack (passive) because of permutation and other computational operations in both algorithms.

B. Performance analysis and system in execution

This sub-section shows the system in the execution (using AnyLogic simulator) and analyze the system performance in terms of time required for Hash computation, operational delay and jitter analysis. Table 6 depicts the chosen parameters for the simulation. More details are presented as follows.

Fig. 9 illustrates the simulation graph of the proposed multi-layered drone-based monitoring system using AnyLogic simulator. In the proposed system, a road network of 1000 km is programmed in a city, and a variation of 60 to 100 vehicles movement per minute with an average speed of 60 km/h is considered. To monitor the traffic, drone movements are random and it is observed that an average of 246 drones per second at layer-1 and 94 drones per second at the layer-2 monitor the traffic in this experimentation. Fig. 10 shows the image processing analysis for measuring the area coverage. The pixelate algorithm can give more precise results for measuring area coverage (shown in blue color). In experimentation, a variation of 5 to 10 block sizes are considered to identify the best

Table 6
The Parameters for Simulation.

Parameter	Value
Simulator	AnyLogic
Number of parallel drones flying areas	6
Drones frequency	60 to 100 per minute
Exit probability	0.3
Road segment	1000 km
Simulation time	24 h
First layer drone altitude	200 m
Second layer drone altitude	300 m
Average speed of drones at top layer	60 km/h (approx.)
Average speed of drones at second layer	40 km/h (approx.)
Top layer's drone communication range	100 to 200 m
Top layer's drone communication range	400 to 500 m
Drone to vehicle detection success rate	>95%
Battery cycle	200 cycle/each

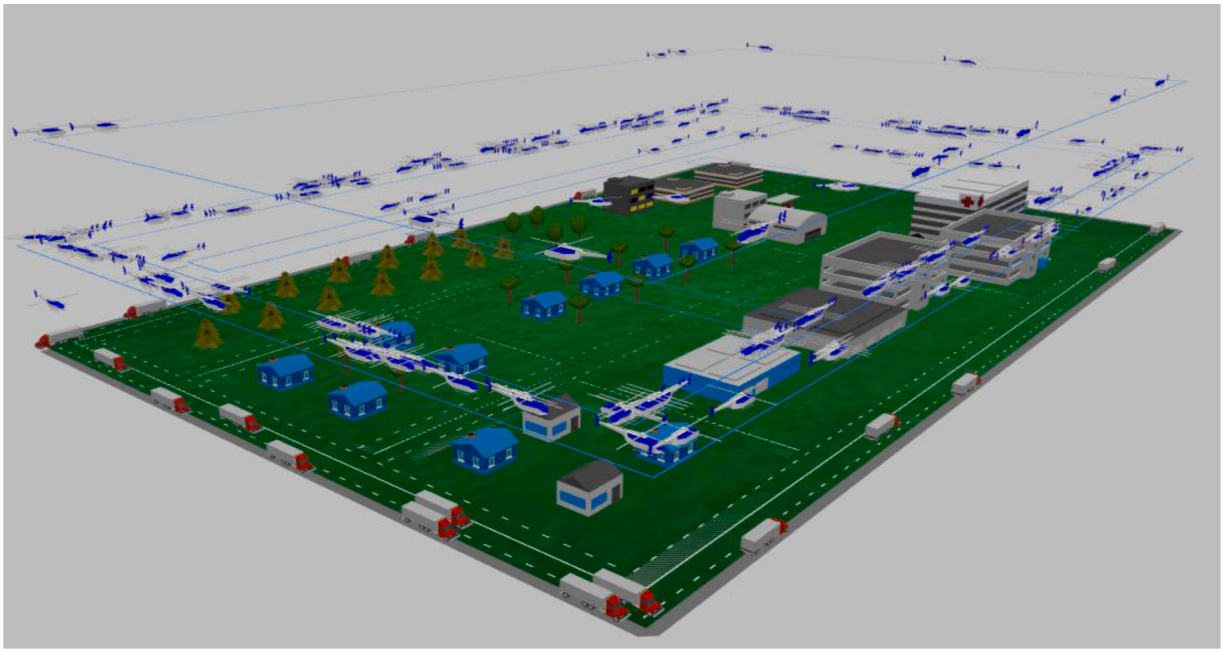


Fig. 9. Multi-layered drone-based simulation (in execution using AnyLogic) for traffic monitoring.

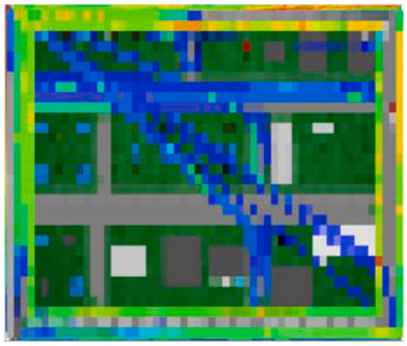


Fig. 10. Image pixelate for precise area coverage monitoring (block size=8).

possible solution for measuring the area coverage. This analysis helps in identification of areas that need to cover for monitoring and traffic record purposes.

MPI-based Hash computational analysis: Fig. 11 shows the variations of time with an increase in the number of hash computation tasks and processors. Results show that the hash computation time varies from 11 s to 39 s approximately for 4 processors, 18 s to 76 s approximately for 3 processors, and 27 s to 115 s approximately for 2 processors. The hash computation time decreases with an increase in processors. This analysis helps in selecting the Hash-computational overhead for drones based on availability of resources. Similar communicational and computational overheads can be counted for drones movements with better QoS. Here, feasibility to integrate multiprocessors with drones is a major challenge and need to be studies in future.

Delay analysis: Fig. 12 illustrates the delay comparison analysis caused by proposed algorithm vs. no security [58] with 150 flying drones. This delay considers processing plus propagation and transmission delays. Results show that algorithm 4 causes more delay compared to algorithm 5 in different scenarios (zoneless drone movement strategy with and without security and multilayer approach [58]) because of multiple use of hashing and other cryptographic primitives. Thus, algorithm 5 is recommended in case of high QoS requirements and algorithm 4 is recommended when comparatively high security is required and implementation of lightweight cryptographic primitives and protocols is possible. The delay analysis in various scenarios based on zone or zoneless strategies is helpful in predicting the future drone movement plans and better QoS scenarios.

Jitter analysis: Jitter is a variation in packet transmission delay. A higher jitter value results in packet loss, drop and/or network congestion. Fig. 13 depicts the jitter comparison analysis in executing the proposed algorithm for 24 h with 150 flying drones. Results demonstrate that jitter varies from 1.5 to 2 msec. for algorithm 4 and 0.7 to 1.3 msec. for algorithm 5. Thus, jitter variation in algorithm 4 is higher than algorithm 5 in different scenarios (zoneless drone movement strategy with and without security and multilayer

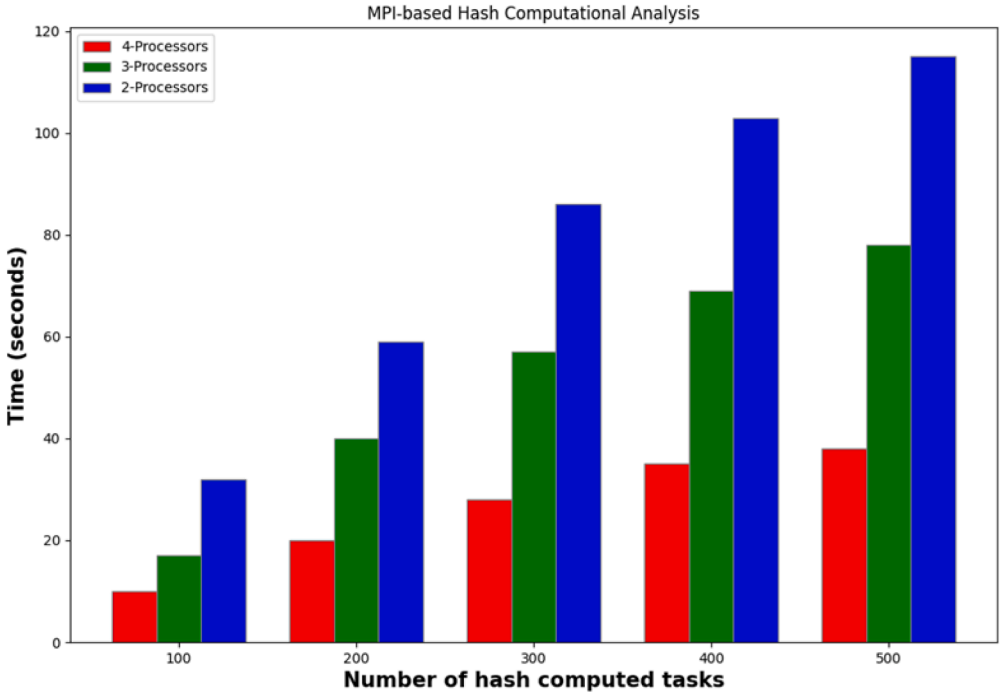


Fig. 11. MPI-based hash computational tasks.

approach [58]) because of the use of hashing and other cryptographic primitives. Table 7 depicts the average jitter comparison analysis of the proposed algorithm vs. state-of-the-art schemes. The results depict that our proposed approaches are more efficient than Nägeli et al. [59] and Sanchez-Aguero et al. [60] because of fast and efficient security mechanisms. Algorithm 5 shows minimum jitter because of lesser computational overhead. In comparison to algorithm 5, algorithm 4 shows higher jitter variations but it provides better security because of the use of security primitives.

Table 8 depicts the temporal communicational cost comparison analysis of the proposed security algorithm (algorithm 4 and algorithm 5) with state-of-the-art approaches. This comparative analysis is drawn for simulation over two system configurations: system-1 (HP Pavillion Gaming 790-0026in i7-8700, 16GB DDR4 RAM, 3.2Ghz, Window OS) and system-2 (HP Z2 Mini Professional Workstation, i7-6700, 8GB DDR4 RAM, 3.4 GHz, Window OS). In results, a linear regression function is observed with input size 'x' in bytes. Results show that algorithm-5 is the most efficient approach because of lightweight mechanisms. Results are comparable with the Model algorithm [61] and Sanchez-Aguero et al. [60] approaches. However, communicational cost analysis of state-of-the-art approaches are higher because of interruptions and heavy primitives and protocols. The computational delay over system-2 is more compared to system-1 because of system configurations.

Table 9 shows the comparative analysis of energy cost analysis of proposed security algorithms with state-of-the-art approaches [60,61]. Results show that energy cost is a linear regression function and proposed algorithm 5 shows minimum energy cost compared with other approaches because of lesser overheads. However, proposed algorithm-1 shows more energy requirements but it provides strong security compared to other approaches as well.

Fig. 14 shows the probability of security interruption variation with change in energy efficiency coefficient for algorithm 5. Energy efficiency coefficient is the ratio of the amount of useful energy offered for security purposes to the amount of energy available in the system. Results show that the probability, with security interruption, reduces with the increase of energy efficiency coefficient. Therefore, the security performance of drone networks with proposed security algorithms can be increased by increasing the energy efficiency of drones in the network. The analysis of energy efficiency with increased security is helpful in planning the drone movements in the zone and zoneless strategies as well. Thus, energy management for drones includes efficient energy generation, energy storage, energy usage, resource optimization, and energy regeneration, which are major phases to consider in futuristic evaluations.

Fig. 15 shows probability of the system's security interruption variations with energy acquisition for algorithm 5. Here, a comparative system security interruption probability variation of proposed SDDN-based approach, which select drones either using (i) maximum drone energy (from top to down layers), (ii) maximum drone energy (from down to top layers), (iii) residual drone energy (from top to down layers), or (iv) residual drone energy (from down to top layer), is made with random selection and model algorithm [61]. Results show that maximum energy and residual energy-based algorithms have lesser security interruption probability compared to random selection or model algorithm. Maximum energy and residual energy-based algorithms show lesser security interruption with variation in energy acquisition coefficient compared to the model algorithm because of lightweight primitives.

Variations in simulation parameters: With the variations in simulation parameters, the following observations are made.

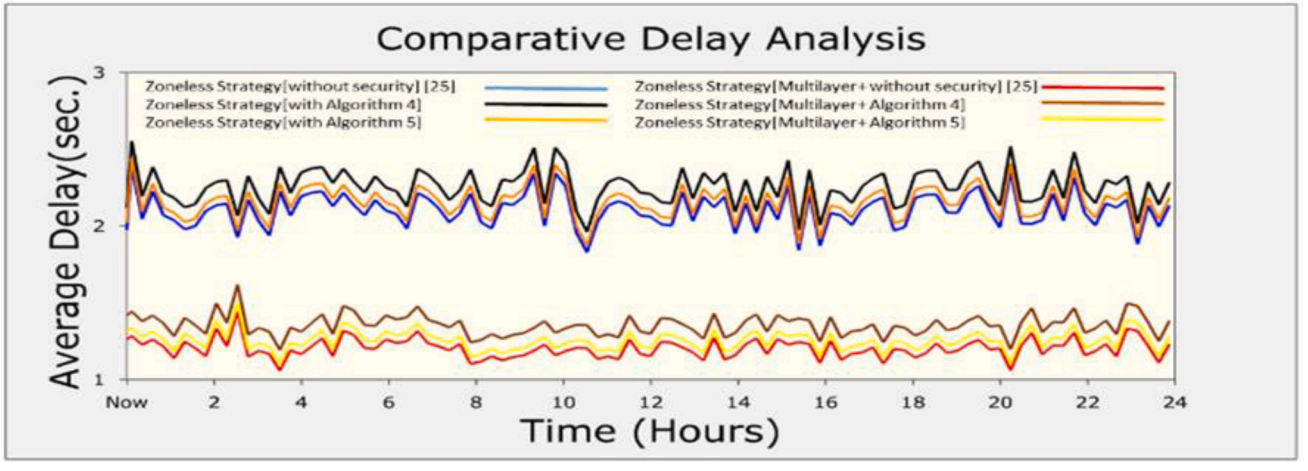


Fig. 12. Comparative average delay analysis in executing proposed algorithms with 150 drones.

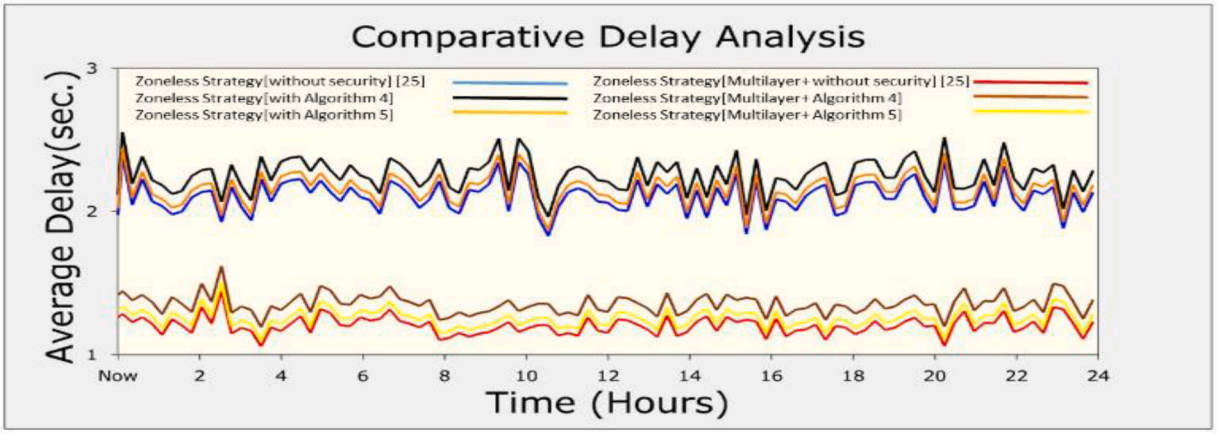


Fig. 13. Comparative jitter analysis in executing proposed algorithms with 150 drones.

Table 7
Comparative average jitter analysis.

Algorithm	Average Jitter (msec.)
Zoneless drone movement Strategy with Algorithm-4**	1.9
Zoneless drone movement Strategy with Algorithm-5**	1.8
Zoneless drone movement Strategy [Multilayer + Algorithm-4**]	1.3
Zoneless drone movement Strategy [Multilayer + Algorithm-5**]	1.1
Nägeli et al. [59] **	2.3
Sanchez-Aguero et al. [60] *	3

* Average analysis of 1-7 drones
** average analysis of upto 150 flying drones

Table 8
Comparative temporal communicational cost analysis of security algorithms.

Algorithm	Time (msec.) System-1	System-2
Proposed Algorithm-4**	1.40+0.4x	1.52+0.3x
Proposed Algorithm-5**	1.25+0.7x	1.30+0.3x
Model Algorithm [61]**	1.54+0.8x	1.68+0.5x
Sanchez-Aguero et al. [60]*	1.81+0.8x	1.93+0.7x

* Average analysis of 1-7 drones
** average analysis of upto 150 flying drones

Table 9
Comparative energy cost analysis of security algorithms.

Algorithm	Energy (mJ) System-1	System-2
Proposed Algorithm-4**	0.30+0.4x	0.12+0.3x
Proposed Algorithm-5**	0.25+0.7x	0.30+0.3x
Model Algorithm [61] **	0.46+0.8x	0.48+0.7x
Sanchez-Aguero et al. [60] *	0.81+0.8x	1.13+0.7x

* Average analysis of 1-7 drones
** average analysis of upto 150 flying drones

- With the change in drone frequency from 60 to 100 per minute to 100 to 150 per minute and 150 to 200 per minute delay in network increases and keeping all other parameters same as shown in Table 5, an increase of 1.9% (minimum) to 3.1% (maximum) increase in delay is observed (for algorithm 4 and algorithm 5) with an increase in drone frequency from 60 to 100 per minute to 100 to 150 per minute. Further, an increase of 2.3% (minimum) to 4.7% (maximum) is observed (for algorithm 4 and algorithm 5) with an increase in drone frequency from 60 to 100 per minute to 150 to 200 per minute. This delay is increased because of

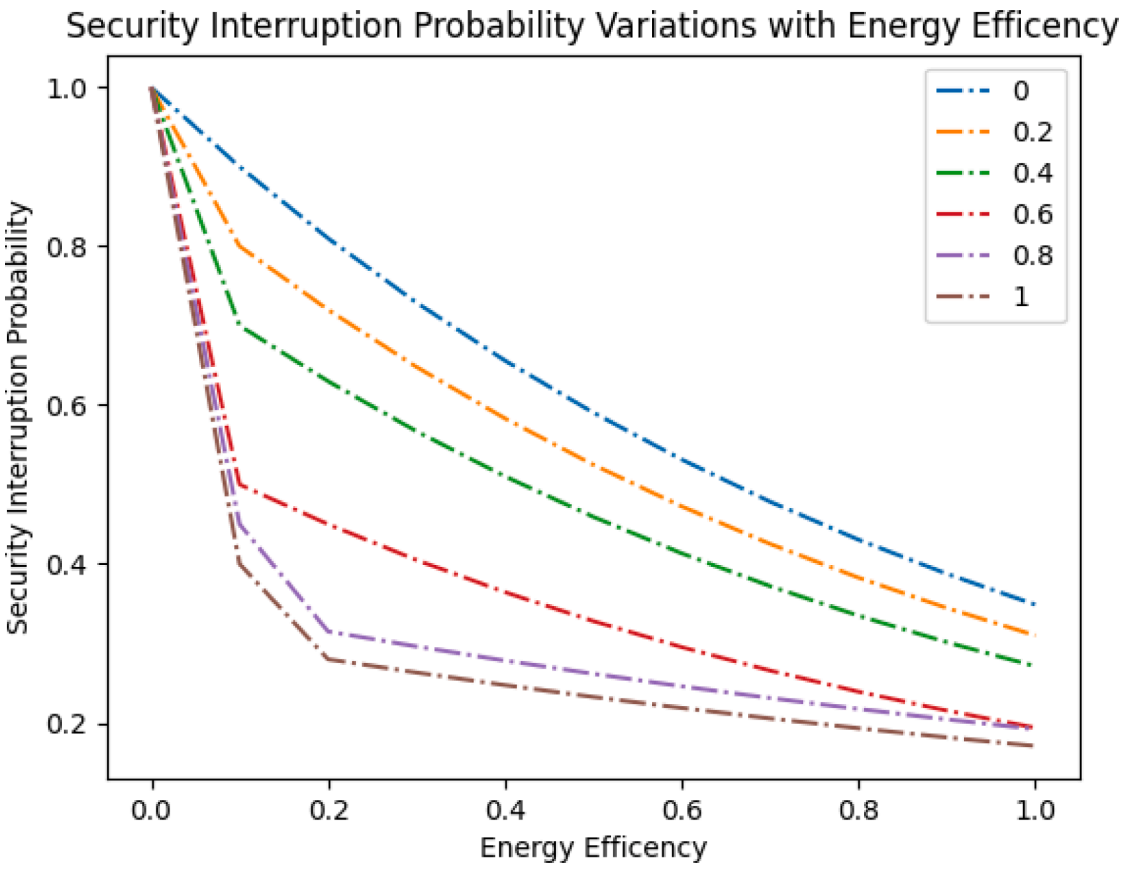


Fig. 14. Security interruption probability variations with change in energy efficiency coefficient for algorithm 5.

additional overhead of increasing the drone in same area which further increases the number of packets in the network. Increase in overhead of number of packets increases the delay.

- With the change in drone frequency from 60 to 100 per minute to 100 to 150 per minute and 150 to 200 per minute delay in network increases and keeping all other parameters same as shown in Table 5, an increase of 0.3% (minimum) to 0.5% (maximum) increase in jitter is observed (for algorithm 4 and algorithm 5) with an increase in drone frequency from 60 to 100 per minute to 100 to 150 per minute. Further, an increase of 0.3% (minimum) to 0.7% (maximum) is observed (for algorithm 4 and algorithm 5) with an increase in drone frequency from 60 to 100 per minute to 150 to 200 per minute. Jitter variation is increases because of increase in number of packets exchanged by additional drones in the network. Large number of drones increases the network overhead which impact on performance especially jitter variations.
- With the change in exit probability from 0.3 to 0.5 and keeping all other parameters the same as shown in Table 5, a decrease in the delay of 0.2% (average of 5 executions) is observed (for algorithm 4 and algorithm 5) for drone frequency of 60 to 100 per minute and 100 to 150 per minute. Likewise, a decrease in the delay of 0.3% (approx..) is observed (for algorithm 4 and algorithm 5) for drone frequency of 100 to 150 per minute. With increase in exit probability, delay is decreased because of either lesser number of drones availability for communication or congestion is reduced in the network.
- With the change in an average speed of drone movement at the top and bottom layers from 60 km/h and 40 km/h to 50 km/h for both layers and keeping all other parameters the same as shown in Table 5, the delay is increased by 0.2% (approx.) (for algorithm 4) and 0.3% (for algorithm 5). Likewise, change in the average speed of drone movement to 40 km/h at both layers (top and bottom) and keeping all other parameters the same as shown in Table 5, the delay is increased to 0.4% approximately (minimum). With decrease in speed to 50 km/h and keeping the inflow of drones same is increasing the number of drones and related communications in the network. Thus, delay is increased.
- With the change in a drone to vehicle detection ratio from >95% to >80%, the average delay is decreased by 0.3% for algorithm 4 and 0.4% for algorithm 5. The average delay is increased because of a large number of packet communication between drones and vehicles to handle failure vehicle detection situations. An increase in delay caused by a decrease in the drone to vehicle detection ratio is because of an increase in the number of failure packets in the network. The increase in failure packets overhead the network which in-turn causes an increase in delays.

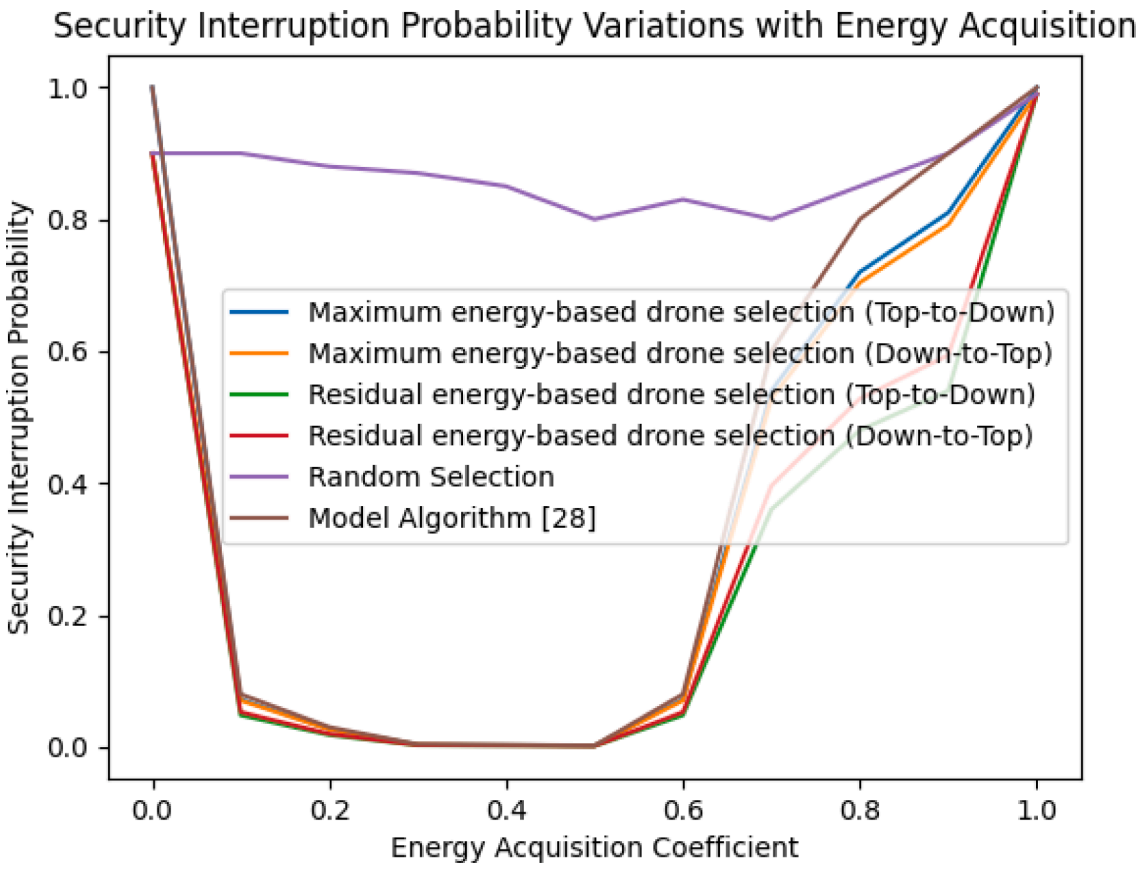


Fig. 15. Comparative security interruption probability variations with change in energy acquisition coefficient for Algorithm 5.

8. Summary, conclusions and future directions

In this paper, drone-based traffic profiling and monitoring architecture is proposed. In this architecture, SDDN framework controls, single and multiple layers-based drone movement and collision avoidance and drones' performance monitoring. Here, image-based real-time data is analyzed for traffic monitoring. The experimentation of image processing is extended to build front, rear and top images of vehicles that are analyzed to identify the vehicles and traffic profiling. Additionally, simulation experimentations are performed to collect sensor-based data and analyze it for traffic engineering. Further, the processing of data is performed in MPI and OpenMP-based parallel and distributed computing (mainly hash computation). Results show that the proposed session key generation and encryption/decryption algorithms have comparatively lesser computational and communicational costs compared to existing approaches. In comparison to alternative systems that include network programming and centralised logical control, the methodology that has been suggested is superior in terms of reliability, scalability, and interoperability. Additionally, a comparative analysis of the SDN-based proposed approach with others (traditional network, cloud/edge/fog-based network and virtualization) shows that the proposed approach is highly flexible, cost-effective, easily separate network control and forwarding functions, open for packet prioritization and blocking, low structural complexity and highly extensible. Strong secrecy, random number processing to avoid disclosure, and protection from the dictionary, brute force, guessing, and false attacks are the strong security primitives of this work. The statistical and formal model (using ProVerif) processes shows that the proposed security algorithms are secure against various attacks because of their properties and use of primitives. The performance analysis shows that OpenMP/MPI-based hash computations are fast and increase with an increase in the number of hash tasks. Security interruption probability variations are lesser for our proposed security algorithms than the existing approaches. Further, the arithmetic operations-based security algorithms consume lesser delay and jitter compared to elliptic curve cryptosystem-based algorithms. -

This work can be extended in various directions. A few of these directions are briefly explained as follows.

- The present work considers a homogenous set of programming and software-defined drones for multi-drone movement and collision avoidance. In the future, a heterogeneous set of drones can be considered for collision avoidance, interoperability and traffic analysis.
- Computing and communication costs for session key generation and encryption/decryption algorithms can be reduced using Quantum Computing with high processing power [62]. In addition, using Quantum Artificial Intelligence (QAI), one of the

advanced AI techniques, anomaly detection with high accuracy and precision will be possible. Thus, the security level of the study will increase.

- The formal model for security verification can be extended to include layer-1 (lightweight cryptographic primitives and protocols) and layer-2 (security mechanisms integrated over layer-1 security solutions) mechanisms to further measure and enhance the system's security levels.
- Performance analysis of proposed work is limited to energy efficiency, jitter, delay and hash-computational performance analysis. However, other QoS parameters including vehicle-to-drone, drone-to-drone, vehicle-to-infrastructure and many more parameters can be considered for evaluation.
- Real-time analysis of multiple homogenous and heterogenous drone network movements, integration with roadside infrastructure and network performance analysis can be performed in future to take the system advantage in traffic engineering.

Declaration of Competing Interest

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

Data Availability

Data will be made available on request.

Acknowledgments

This work is supported in part by the Lancaster University GCRF SEED CORN funding with grant agreement UTS1000XS14. We thank Prof. Helen Karatza (Editor-in-Chief), associate editor and anonymous reviewers for their constructive suggestions and guidance on improving the content and quality of this paper.

References

[1] R. Cucchiara, M. Piccardi, P. Mello, Image analysis and rule-based reasoning for a traffic monitoring system, IEEE Conf. Intell. Transp. Syst. , ITSC 1 (2) (1999) 758–763.
[2] Z. Liu, S. Jiang, P. Zhou, M. Li, A participatory urban traffic monitoring system: the power of bus riders, IEEE Trans. Intell. Transp. Syst. 18 (10) (2017) 2851–2864.
[3] S. Lee, J. Zhong, Z. Kim, K. Dimitrijevic, B. Du, B. Gutesa, Examining the applicability of small quadcopter drone for traf-fic surveillance and roadway incident monitoring, Proceedings of the Transportation Research Board 94th Annual Meeting 15–4184 (2015) 15.
[4] C. Liu, G. Zhang, B. Li, R. Ma, D. Jiang, Y. Zhao, A SDN-based intelligent prediction approach to power traffic identification and monitoring for smart network access, Wirel. Netw. 27 (5) (2021) 3665–3676.
[5] S. Kim, S. Yoon, H. Lim, Deep reinforcement learning-based traffic sampling for multiple traffic analyzers on software-defined networks, IEEE Access 9 (2021) 47815–47827.
[6] Y. Yang, A SDN-based traffic estimation approach in the internet of vehicles, Wirel. Netw. (2021) 1–12.
[7] I. Yaqoob, I. Ahmad, E. Ahmed, A. Gani, M. Imran, N. Guizani, Overcoming the key challenges to establishing vehicular communication: is SDN the answer? IEEE Commun. Mag. 55 (7) (2017) 128–134.
[8] I. Aliyu, M.C. Feliciano, S. Van Engelenburg, D.O. Kim, C.G. Lim, A blockchain-based federated forest for SDN-enabled in-vehicle network intrusion detection system, IEEE Access 9 (2021) 102593–102608.
[9] M. Zhou, L. Han, H. Lu, C. Fu, Y. Qian, Attack detection based on invariant state set for SDN-enabled vehicle platoon control system, Veh. Commun. (2021) 100417.
[10] M. Taneja, N. Garg, Smart traffic monitoring and alert system using VANET and deep learning, in: Proceedings of the International Conference on Innovative Computing and Communications, Springer, Singapore, 2022, pp. 525–536.
[11] T.M. Dang, M.L. Kieu, Tramon: An Automated Traffic Monitoring System for High Density, Mixed and Lane-Free Traffic, Mixed and Lane-Free Traffic (2022).
[12] S. Lee, S.H. Park, Concept drift modeling for robust autonomous vehicle control systems in time-varying traffic environments, Expert Syst. Appl. 190 (2022), 116206.
[13] X. Shi, C. Yang, W. Xie, C. Liang, Z. Shi, J. Chen, Anti-drone system with multiple surveillance technologies: architecture, implementation, and challenges, IEEE Commun. Mag. 56 (4) (2018) 68–74.
[14] M.A. Khan, W. Ectors, T. Bellemans, D. Janssens, G. Wets, UAV-based traffic analysis: a universal guiding framework based on literature survey, Transp. Res. Procedia 22 (2016) (2017) 541–550.
[15] A. De Bruin, M.J. Booysen, Drone-Based Traffic Flow Estimation and Tracking Using Computer Vision, South African Transp. Conf. Proj. Intell. Transp. Syst. (2015) 869–878.
[16] H. Niu, N. Gonzalez-Prelcic, R.W. Heath, A UAV-Based Traffic Monitoring System - Invited Paper, in: 2018 IEEE 87th Vehicular Technology Conference (VTC Spring), 2018, pp. 1–5.
[17] P. Garcia-Aunon, J.J. Roldán, A. Barrientos, Monitoring traffic in future cities with aerial swarms: Developing and optimizing a behavior-based surveillance algorithm, Cogn. Syst. Res. 54 (2019) 273–286.
[18] Y. Altshuler, A. Pentland, A.M. Bruckstein, Optimal dynamic coverage infrastructure for large-scale fleets of reconnaissance UAVs," in Swarms and Network Intelligence in Search, Springer, 2018, pp. 207–238.
[19] S.A.R. Naqvi, S.A. Hassan, H. Pervaiz, Q. Ni, Drone-Aided Communication as a Key Enabler for 5G and Resilient Public Safety Networks, IEEE Commun. Mag. 56 (1) (2018) 36–42.
[20] Y. Liu, K. Xiong, Q. Ni, P. Fan, K. Ben Letaief, UAV-Assisted Wireless Powered Cooperative Mobile Edge Computing: Joint Offloading, CPU Control, and Trajectory Optimization, IEEE Internet Things J 7 (4) (2020) 2777–2790.
[21] C. Christodoulou, P. Kolios, Optimized tour planning for drone-based urban traffic monitoring, in: 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), 2020, pp. 1–5.
[22] N.A. Khan, N.Z. Jhanjhi, S.N. Brohi, R.S.A. Usmani, A. Nayyar, Smart traffic monitoring system using Unmanned Aerial Vehicles (UAVs, Comput. Commun. 157 (January) (2020) 434–443.

[23] V. Balasubramanian, S. Otoum, M. Aloqaily, I. Al Ridhawi, Y. Jararweh, Low-latency vehicular edge: A vehicular infrastructure model for 5G, Simulation Modelling Practice and Theory 98 (2020), 101968.
[24] T.G. Nguyen, T.V. Phan, D.T. Hoang, T.N. Nguyen, C. So-In, Federated deep reinforcement learning for traffic monitoring in SDN-based IoT networks, IEEE Transactions on Cognitive Communications and Networking 7 (4) (2021) 1048–1065.
[25] C. Kyrkou, T. Theocharides, EmergencyNet: Efficient Aerial Image Classification for Drone-Based Emergency Monitoring Using Atrous Convolutional Feature Fusion, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 13 (2020) 1687–1699.
[26] E. Barmpounakis, N. Geroliminis, On the new era of urban traffic monitoring with massive drone data: The pNEUMA large-scale field experiment, Transp. Res. Part C Emerg. Technol. 111 (July 2019) 50–71, 2020.
[27] S.S.C. Congress, A.J. Puppala, A. Banerjee, U.D. Patil, Identifying hazardous obstructions within an intersection using unmanned aerial data analysis, International Journal of Transportation Science and Technology 10 (1) (2021) 34–48.
[28] M. Hamurcu, E.R.E.N. Tamer, Selection and Ranking of the Most Suitable Drones for Sustainable Traffic Management Using Multi-Criteria Analysis Approach (2021).
[29] R. Guirado, J.C. Padró, A. Zoroa, J. Olivert, A. Bukva, P. Cavestany, StratoTrans: Unmanned Aerial System (UAS) 4G Communication Framework Applied on the Monitoring of Road Traffic and Linear Infrastructure, Drones 5 (1) (2021) 10.
[30] A. Kumar, S. Jain, Drone-Based Monitoring and Redirecting System. Development and Future of Internet of Drones (IoD): Insights, Trends and Road Ahead, Springer, Cham, 2021, pp. 163–183.
[31] D. Basu, S. Kal, U. Ghosh, R. Datta, SoftDrone: Softwarized 5G assisted drone networks for dynamic resource sharing using machine learning techniques, Computers and Electrical Engineering 101 (2022), 107962.
[32] E.V. Butilă, R.G. Boboc, Urban Traffic Monitoring and Analysis Using Unmanned Aerial Vehicles (UAVs): A Systematic Literature Review, Remote Sensing 14 (3) (2022) 620.
[33] T. Nguyen Gia, J.P. Queralta, T. Westerlund, Exploiting LoRa, edge, and fog computing for traffic monitoring in smart cities, INC (2020).
[34] A. Beg, A.R. Qureshi, T. Sheltami, A. Yasar, UAV-enabled intelligent traffic policing and emergency response handling system for the smart city, Pers. Ubiquitous Comput. (February) (2020).
[35] V. Balasubramanian, S. Otoum, M. Reisslein, VeNet: Hybrid Stacked Autoencoder Learning for Cooperative Edge Intelligence in IoV, IEEE Transactions on Intelligent Transportation Systems (2022).
[36] G. Bathla, K. Bhadane, R.K. Singh, R. Kumar, R. Aluvalu, R. Krishnamurthi, A. Kumar, R.N. Thakur, S. Basheer, Autonomous Vehicles and Intelligent Automation: Applications, Challenges, and Opportunities, Mobile Information Systems 2022 (2022).
[37] F. Xhafa, Autonomous and Connected Heavy Vehicle Technology, Academic Press, 2022.
[38] P. Srikanth, A. Kumar, Automatic vehicle number plate detection and recognition systems: Survey and implementation. Autonomous and Connected Heavy Vehicle Technology, Academic Press, 2022, pp. 125–139.
[39] D.P. Isravel, S. Silas, E.B. Rajsingh, Long-term traffic flow prediction using multivariate SSA forecasting in SDN based networks, Pervasive and Mobile Computing 83 (2022), 101590.
[40] J. Ma, R. Jin, L. Dong, G. Zhu, X. Jiang, Implementation of SDN traffic monitoring based on Ryu controller, International Symposium on Computer Applications and Information Systems (ISCAIS 2022) 12250 (2022, May) 203–212.
[41] I. Rabet, S.P. Selvaraju, H. Fotouhi, M. Alves, M. Vahabi, A. Balador, M. Björkman, SDMob: SDN-Based Mobility Management for IoT Networks, Journal of Sensor and Actuator Networks 11 (1) (2022) 8.
[42] S. Lahlou, Y. Moukafih, A. Sebbar, K. Zkik, M. Boulmalf, M. Ghogho, TD-RA policy-enforcement framework for an SDN-based IoT architecture, Journal of Network and Computer Applications 204 (2022), 103390.
[43] H. Polat, M. Türkoğlu, O. Polat, A. Şengür, A novel approach for accurate detection of the DDoS attacks in SDN-based SCADA systems based on deep recurrent neural networks, Expert Systems with Applications 197 (2022), 116748.
[44] J. Sun, T. Wo, X. Liu, R. Cheng, X. Mou, X. Guo, H. Cai, R. Buyya, CloudSimSFC: Simulating service function chains in Multi-Domain Service Networks, Simulation Modelling Practice and Theory (2022), 102597.
[45] B. Alhijawi, S. Almajali, H. Elgala, H.B. Salameh, M. Ayyash, A survey on DoS/DDoS mitigation techniques in SDNs: Classification, comparison, solutions, testing tools and datasets, Computers and Electrical Engineering 99 (2022), 107706.
[46] A. Kumar, Jesus de, D.A. Pacheco, K. Kaushik, J.J. Rodrigues, Futuristic view of the Internet of Quantum Drones: Review, challenges and research agenda, Vehicular Communications (2022), 100487.
[47] A. Kumar, S. Bhatia, K. Kaushik, S.M. Gandhi, S.G. Devi, A.D.J. Diego, A. Mashat, Survey of promising technologies for quantum drones and networks, IEEE Access 9 (2021) 125868–125911.
[48] S. Verma, A. Soni, V. Mishra, V. Gupta, R. Krishnamurthi, A. Kumar, Smart automated system for classification of emergency heavy vehicles and traffic light controlling. Autonomous and Connected Heavy Vehicle Technology, Academic Press, 2022, pp. 245–262.
[49] G. Sharma, A. Kumar, S.S. Gill, Applications of blockchain in automated heavy vehicles: Yesterday, today, and tomorrow. Autonomous and Connected Heavy Vehicle Technology, Academic Press, 2022, pp. 81–93.
[50] S. Jain, N.J. Ahuja, P. Srikanth, K.V. Bhadane, B. Nagaiah, A. Kumar, C. Konstantinou, Blockchain and Autonomous Vehicles: Recent Advances and Future Directions, IEEE Access, 2021.
[51] S. Jain, A. Kumar, K. Kaushik, R. Krishnamurthi, Autonomous driving systems and experiences: A comprehensive survey, Autonomous and Connected Heavy Vehicle Technology (2022) 65–80.
[52] J. Son, R. Buyya, A taxonomy of software-defined networking (SDN)-enabled cloud computing, ACM Comput. Surv. 51 (3) (2018).
[53] J. Singh, A. Gimekar, S. Venkatesan, An efficient lightweight authentication scheme for human-centered industrial Internet of Things, Int. J. Commun. Syst. (July) (2019) 1–13.
[54] Y. Zhang, D. He, L. Li, B. Chen, A lightweight authentication and key agreement scheme for Internet of Drones, Comput. Commun. 154 (January) (2020) 455–464.
[55] ProVerif: Cryptographic protocol verifier in the formal model (September 26, 2020). https://prosecco.gforge.inria.fr/personal/bblanche/proverif/ [Last].
[56] AnyLogic (September 26, 2020). https://www.anylogic.com/ [Last Accessed:].
[57] JaamSim (September 26, 2020). https://jaamsim.com/ [Last Accessed:].
[58] A. Kumar, R. Krishnamurthi, A. Nayyar, A.K. Luhach, M.S. Khan, A. Singh, A novel Software-Defined Drone Network (SDDN)-based collision avoidance strategies for on-road traffic monitoring and management, Veh. Commun. (2020), 100313.
[59] T. Nageli, L. Meier, A. Domahidi, J. Alonso-Mora, O. Hilliges, Real-time planning for automated multi-view drone cinematography, ACM Trans. Graph. 36 (4) (2017).
[60] V. Sanchez-Aguero, F. Valera, B. Nogales, L.F. Gonzalez, I. Vidal, VENUE: Virtualized Environment for Multi-UAV Network Emulation, IEEE Access 7 (2019) 154659–154671.
[61] Z. Lv, The security of Internet of drones, Comput. Commun. 148 (September) (2019) 208–214.
[62] S.S. Gill, et al., AI for Next Generation Computing: Emerging Trends and Future Directions, Internet of Things 19 (2022) 1–34.