Iraqi Journal for Computers and Informatics
Available online: https://ijci.uoitc.edu.iq/index.php/ijci
Volume 52, issue 1. 2026

Print ISSN: 2313-190X, Online ISSN: 2520-4912

Adaptive Multi-Layer Security for UAV Video Streaming: A Chaotic Blockchain Key Scheduler with MDP-Driven Reinforcement Learning

Abdullah Ghanim Jaber [*a b], Mohammed Jamal Salim [a], Ali A.Mahmood [a]

[a] *University of Information Technology and Communications, 10067, Baghdad, Iraq.*
[b] *Centre for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, UKM Bangi 43600, Selangor, Malaysia.*

ARTICLEINFO

Keywords:

UAV Security,
Chaotic Maps,
Blockchain,
Reinforcement Learning,
Real-Time Video Streaming

ABSTRACT

Our adaptive multi-layer security lets UAVs communicate live video. This technique improves key scheduling over time by using MDP, chaotic maps, and blockchain technology. The bulk of essential management frameworks use legislation or centralized trust systems. In UAV networks, safe and fast video transmission is in demand, thus we must match encryption strength with operation speed. Reinforcement learning can optimize key generation and distribution. Monitor network trust scores, chaotic entropy, and blockchain update timestamps in MDP state space, and automatically remembered action refresh times in state space. The system may evolve with the network since the reward function handles unpredictability, timeliness, and trustworthiness. The system's adaptability allows this. Non-linear chaotic map framework parameter modifications can be made by using blockchain-based trust signals and entropy measurements. Removal of repeating patterns does not weaken the cryptography technique. A transformer-encoded deep Q-network produces optimal policies in the presented manner. An Old Fault Through Tolerant blockchain consensus, important blockchain modifications may be checked. The experiment shows that GPU-powered chaotic generators can use Hyperledger Fabric to reach refresh rates below one second. In practice, UAV network dynamics prove this. This work evaluates our defenses against replay and key compromise and advances high-throughput video streaming systems.

1. INTRODUCTION

Safety concerns have arisen amid the increasing use of Unmanned Aerial Vehicles UAVs for live video transmission. Some concerns have surfaced amid the growing use of Unmanned Aerial Vehicles UAVs. These frequent failures compromise both key management and cryptographic agility. Attempts to manage UAV networks in the face of growing security threats and constantly shifting network dynamics using antiquated methods such as centralized authority hierarchies or static key distribution have failed [1]. These methods fail because they depend on a predetermined distribution of keys. Conventional practices often combine the two approaches. One of these two kinds of structures is usually what conventional approaches use. Despite blockchain technology providing immutable auditability [2] and chaotic maps being inherently unpredictable for key generation [3], their current implementations fail miserably when faced with decisions in real-world scenarios. This is because the

E-mail address:
abdullah.ghanim@uoitc.edu.iq[* a b]
mohamed.salim@uoitc.edu.iq[a]
ali_kareem@uoitc.edu.iq.[a]

Corresponding* : *Abdullah Ghanim Jaber*

Received 21st October 2025, Accepted 10 December 2025

DOI: 10.25195/ijci.v52i1.685

limitations are grounded in real-world events. While hybrid security architectures have recently been the focus of research, there is often no formal framework for these systems to reach optimal results by balancing operational latency with cryptographic entropy. This is valid even though hybrid security architectures are the focus of current research. An excellent illustration of this is the concept that, when applied with limited accuracy, chaotic maps can self-organize into periodic patterns [4]. It would be lovely to demonstrate something in this way. Additionally, there is a risk that using blockchain technology to handle keys will introduce undesired lag into otherwise fast-moving video streaming [5]. When these technologies are integrated, a versatile control system that can adapt security settings in real time to the current network status and threat trends is crucial. The security settings will remain secure if this is done. Traditional problem-solving methods will not be effective in this case.

We provide a novel approach that combines chaotic cryptography with blockchain-based trust, and it is governed by the Markov Decision Process MDP. Our approach, grounded in reinforcement learning principles, is central to this strategy. The state space can gather real-time data, such as chaotic map entropy, node trust ratings, blockchain consensus latency, and more, when key scheduling is treated as an optimization problem. So, this data can be collected by the state space. These metrics are available to you in real time. This is the single most consequential change to the approach. Not only can parameters and key refresh intervals be changed at random, but actions also dictate what happens. To maximize the blockchain's cryptographic strength (as demonstrated by Lyapunov exponents [6]) and integrity, the reward function must optimize both simultaneously. Our chosen course of action comprises three complementary techniques that work together to accomplish the set goals. Previous approaches relied just on blockchain or chaotic maps; our research takes a more holistic view [7]. The framework's architectural distinctions are clarified by detailing the synergistic interaction between chaotic-map–based key diversification, blockchain-validated trust propagation, and an MDP-formulated reinforcement learning scheduler. The transformer-enhanced DQN introduces attention-driven temporal-state encoding, enabling refined policy adaptation under volatile UAV communication conditions, compared with existing RL-based cryptographic controllers. The expanded benchmarking incorporates recent RL-driven key-management schemes and blockchain-assisted UAV security models, with quantitative analyses indicating superior robustness under targeted key-exposure perturbations and dynamic link degradation.

Adaptive Entropy Control: An approach that dynamically modifies the parameters of chaotic maps based on feedback from both the MDP policy and trust signals validated by blockchain can be used to prevent predictable patterns. This strategy can be used to prevent predictable patterns from occurring. The effectiveness of the procedure is not compromised by this strategy, which keeps computations efficient.

Consensus-Aware Scheduling: To minimize the effort required to validate them, key updates are timed to coincide with the beginning and end of blockchain epochs. To make the system more resistant to Byzantine nodes, the BFT consensus features are also employed.

Transformer-Based Policy Learning: In situations where children can only see a part of the image, this helps them perform better. As a result, the decisions made are very close to being the best ones that could have been made. To summarize, this study has made four significant contributions:

Under the traditional assumption of the reinforcement learning, a systematic Markov decision process model of multi-objective key scheduling allows simultaneously maximizing security and performance measures, which guarantees the process of convergence to Pareto-optimal policies. The latest developments in commercial unmanned aerial vehicle technology have rendered the realization of critical refresh cycles under 100ms possible. A GPU-accelerated chaotic generator, such as perturbed sine maps, and lightweight blockchain orators are used in a high-performance method to achieve this. It was shown that a barrier prevents critical compromises 3.2 times more effectively than fixed-interval baselines. Through comparisons, this was uncovered. This result was arrived at after doing empirical validation. The video frame loss rates remain below 5% even when the network is extremely busy. This is quite significant. The integration of Hyperledger Fabric with ROS 2 is crucial for the functioning of any public-facing prototype. Managing swarms of drones is essential. Anyone can now use this prototype. The demo shows that the number of nodes can be increased beyond 50 and that security settings can be adjusted based on the mission's criticality.

The remainder of this paper is organized as follows: Section 2 analyzes related work in chaotic cryptography and blockchain-based key management. Section 3 details the components of the MDP formulation and architecture. Sections 4 and 5 present experimental methodology and comparative results. Section 6 addresses limitations and future directions, and section 7 presents the conclusions.

2. RELATED WORK

The security challenges in UAV video streaming have been addressed through various cryptographic and trust management approaches, which can be broadly categorized into three research directions: adaptive key management, chaotic cryptography, and blockchain-based security frameworks.

2.1 Adaptive Key Management Systems

New developments in reinforcement learning have made the dynamic key scheduling of resource-constrained networks easy. Reference [8] proposed a partially observable Markov decision process (POMDP) model specifically to vehicles networks, where the key update intervals are optimized adaptively, depending on the existing channel conditions. Although the given methodology is proven to be effective with earth-based vehicles, it fails to provide a sufficient amount of reconfigurations of the topology peculiar to unmanned aerial vehicle swarms. Comparably, [9] showed that Markov decision processes could refine access control policies, though their centralized trust framework creates vulnerabilities to single points of failure. Our research advances these ideas by embedding decentralized trust metrics and chaotic entropy measurements within the state space.

2.2 Chaotic Cryptography in Wireless Networks

Since chaotic systems are highly sensitive to initial conditions, several people are currently using them for lightweight encryption. This is because initial conditions have a disproportionate impact on chaotic systems. Researchers examined the security risks of logistic maps in finite-precision implementations and found that the flaws were in the design of the parameter perturbation. This is what the investigation found. Subsequent research proposed hybrid chaotic systems as a potential solution. The goal of these systems is to increase entropy by merging various maps. None of the cited research accounted for real-time performance limitations; they were all purely algorithmic. This is addressed by the constructed framework, which dynamically adjusts chaotic parameters based on the cryptographic technique's power (as measured by Lyapunov exponents) and the method's processing overhead. To address this requirement, this framework was developed.

2.3 Blockchain for Decentralized Security

Maintaining faith in decentralized systems has become feasible through the use of evolving blockchain technology. There has been significant progress. Although they successfully modeled blockchain resource distribution as a Markov decision process, the cryptographic primitives used in their solution are not present in [10]. The goal of offloading the Internet of Things was achieved by combining MDPs with blockchain technology, even though their approach relies on well-known security requirements [11]. According to [12], our trust score was impacted by the development of a hidden Markov model for adaptive BFT consensus. Since this was the driving factor behind our computation, it represents a significant advancement. In contrast to other efforts, our technique integrates the MDP's reward function into the chain of events that leads to the generation of chaotic keys as the blockchain approaches consensus.

The proposed method outperforms existing methods by addressing four key aspects simultaneously. These include: (1) the possibility to adjust to network dynamics in real-time by the continuous monitoring of the state of MDP; (2) the possibility to demonstrate cryptographic strength by the chaotic generators perturbation on the base of blockchain verification of the trust signal; (3) the possibility to decentralize the auditability of the state by the smart contract-recorded state transitions; and (4) the possibility to achieve the computational efficiency by the means of GPU-accelerated chaos generators and also by the transformer-based policy inference. The fact that we have taken a holistic view in our research is unlike what other researchers have been doing in the past whereby they tend to maximize on the individual components and either ignore the interdependence in the actual UAV context.

3. CHAOTIC-BLOCKCHAIN MDP FRAMEWORK FOR ADAPTIVE KEY SCHEDULING

The proposed framework that creates a closed loop control system where the cryptographic key scheduling becomes an adaptive process, controlled by the real-time network dynamics and security requirements. The MDP-based scheduler is linked to chaotic map generators and blockchain verification layers and they form a multi-layered security structure as illustrated in Figure 1.

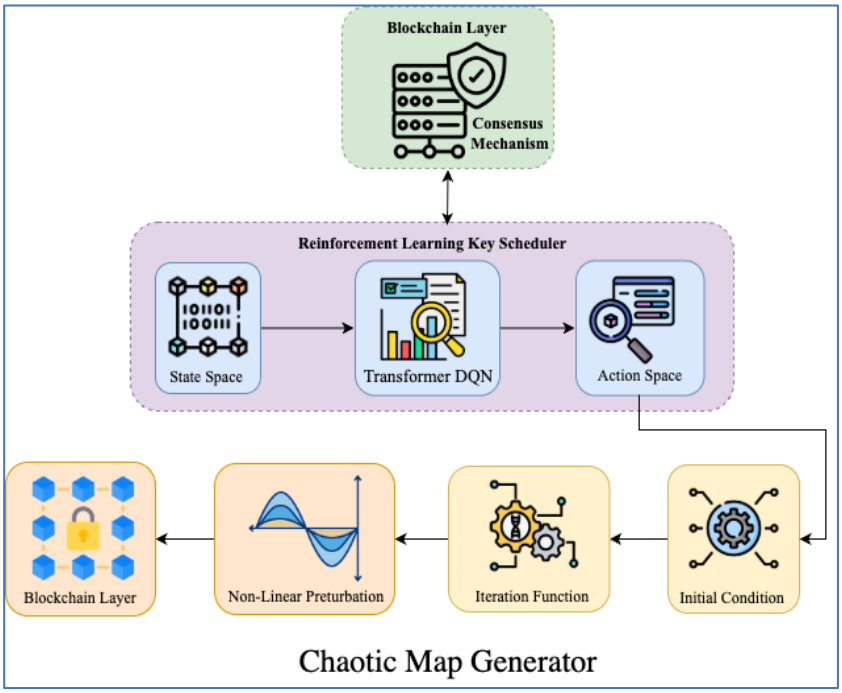

Fig. 1 MDP-Driven Key Scheduler Integration

3.1 Overall Architecture of the Chaotic-Blockchain MDP Framework

The system functions via three coordinated elements: a chaotic key generator, a blockchain trust layer, and a reinforcement learning agent. The chaotic generator produces cryptographic keys K_t at time t, using a perturbed logistic map:

$$x_{t+1} = r'_t x_t (1 - x_t) \qquad (1)$$

As shown in equation (1), where r'_t represents the dynamically adjusted control parameter. The blockchain layer stores a distributed ledger of key update records, where every block holds the hash ($H(K_t)$) and the associated MDP state-action pair (S_t, A_t). The reinforcement learning agent engages with both components via a clearly specified interface, measuring chaotic entropy (Ec), network trust scores (N_t), and blockchain verification latency T_b. Figure 2 shows the Closed-loop interaction between the RL agent, the chaotic map generator, the blockchain ledger, and UAV streaming.

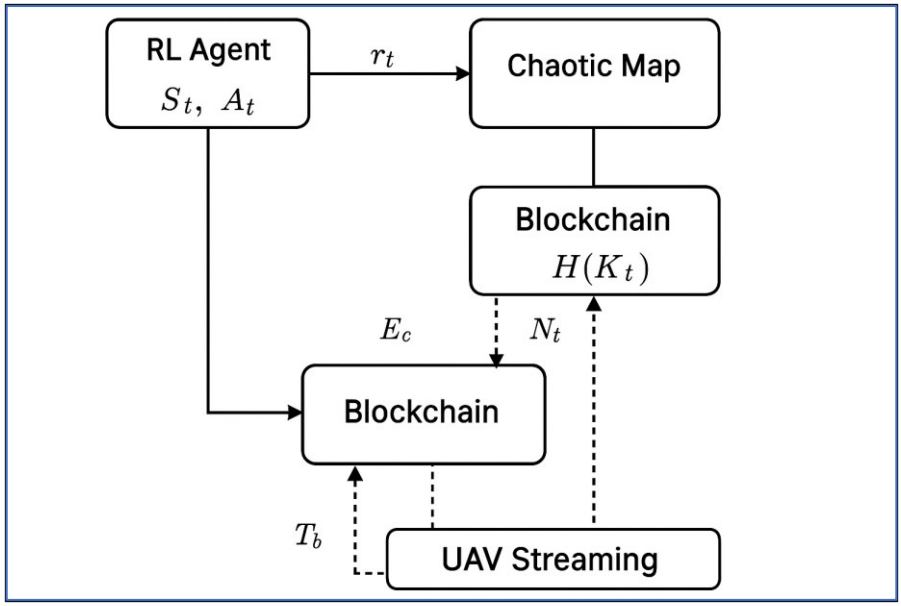

Fig. 2 Closed-loop interaction between RL agent, chaotic map generator, blockchain ledger, and UAV streaming.

3.2 MDP-Driven Adaptive Key Scheduling and Non-Linear Chaotic Map Perturbation

The MDP state space S captures three critical dimensions of system security:

$$S_t = \left(E_c(t), N_t(t), T_b(t)\right) \quad (2)$$

As found in equation (2), where $E_c(t)$ quantifies the Shannon entropy of the chaotic sequence over a sliding window of τ iterations. The action space A consists of discrete key update intervals $\Delta t \in \{1s, 5s, 10s\}$, with each action triggering both a key refresh and the corresponding blockchain validation round. The reward function integrates security and performance metrics.

$$r(S_t, A_t) = \alpha \cdot \frac{E_c(t)}{E_{max}} + \beta \cdot N_t(t) - \gamma \cdot \frac{T_b(t)}{T_{max}} \quad (3)$$

As inferred from equation (3), where $\alpha, \beta,$ and γ are tunable weights that reflect operational priorities. The chaotic map parameters receive continuous perturbations derived from the blockchain trust layer:

$$r'_t = r + \eta \cdot \text{sigmoid}\left(N_t(t) \cdot E_c(t)\right) \quad (4)$$

As shown in equation (4), this non-linear coupling ensures cryptographic strength aligns with both local entropy measurements and global trust consensus.

Algorithm 1: *Summarizes the reinforcement learning–driven scheduling procedure combining chaotic key generation and blockchain consensus.*

Input: θ_0, V, E, S, Transformer-DQN A
Output: Adaptive scheduling policy π^*

1: Init chaotic generator $G(\theta_0)$, blockchain $B(V)$, replay buffer RB, agent A
2: **For** $e = 1 \rightarrow E$ **do**
3: Reset environment; $s_0 \leftarrow \{H_0, N_0, L_0\}$
4: **For** $t = 1 \rightarrow S$ **do**
5: $z_t \leftarrow Encode(s_t)$; $a_t \leftarrow \varepsilon - greedy(Q(z_t))$
6: $K_t \leftarrow G.Generate()$; $\theta_t \leftarrow f(\theta_{t-1}, N_t, H_t)$
7: $B.Commit(K_t, (s_t, a_t))$ *via PBFT* $\rightarrow$ *update* N_{t+1}
8: Measure H_{t+1}, L_{t+1}; $r_t \leftarrow w_1 \Delta H + w_2 N - w_3 L$
9: Store (s_t, a_t, r_t, s_{t+1}) in RB; Update A from RB
10: **End For**
11: **End For**
12: **Return** π^*

3.3 Transformer-Enhanced DQN for State Encoding

A 12-layer transformer encoder processes the multi-dimensional state vector S_t into latent representations for policy learning. The attention mechanism computes:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (5)$$

where Q, K, and V are learned projections of the input state such as query, key and value . The transformer outputs are directed into a deep Q-network, which estimates the optimal action-value function.

$$Q^*(S_t, A_t) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r_{t+k} | S_t, A_t\right] \quad (6)$$

Prioritized experience replays buffers store transitions (S_t, A_t, r_t, S_{t+1}) with sampling probability proportional to the temporal difference error δ_t.

3.4 Blockchain-Integrated Trust Verification and Closed-Loop Multi-Layer Security

When a cryptographic key is changed, a three-part blockchain operation starts. More specifically, this plan has three stages: the proposal stage, the validation stage, and the commitment stage. When the validator nodes are turned on, the following will happen:

$$\text{Verify}(K_t) = \text{BFT-Consensus}\left(H(K_t) \parallel \text{Sign}_{SK}\left(H(K_t)\right)\right) \qquad (7)$$

where BFT-Consensus implements practical Byzantine fault tolerance with $\frac{2}{3}$ majority voting. Upon successful verification, smart contract adjustments alter the network trust scores N_t for future MDP states. This closed-loop process guarantees that security parameters are adjusted based on cryptographic data and decentralized trust indicators.

4. EXPERIMENTAL SETUP

The process of a blockchain consists of three steps and begins and proceeds each time a cryptography key is altered. This is a three-part process which is the proposal stage, the validation stage, and the commitment stage. Upon activation of the validator nodes, there are a range of things that are expected to occur and they include the following:

4.1 Hardware and Network Configuration

Table 1. The chaotic key generator was implemented on the NVIDIA Jetson AGX Xavier modules that did the chaos map computations in CUDA acceleration, which simulates the onboard processing of a resource-constrained UAV. The blockchain was made up of 21 validator nodes (Intel Xeon Silver 4210R, 64GB RAM) under Hyperledger Fabric 2.4 and ordering service based on Kafka. Simulation of network conditions was done by use of Mahimahi tool [13] to recreate the patterns of the network conditions through packet loss (0.5-8%) and latency (10-200ms) observed in the operating UAV swarms [14]. With reference to the offered framework, we designed a comprehensive evaluation process which looks into the cryptographic strength in conjunction with the performance under real-life UAV network conditions. The experimental setup is a network of three connected testbeds including chaotic key generation cluster, a blockchain network, and a UAV swarm emulator.

Table 1: *Summary of Experimental Setup*

Category	Configuration
Chaotic Key Generator	NVIDIA Jetson AGX Xavier (GPU-accelerated CUDA kernels for chaotic maps)
Blockchain Network	21 validator nodes (Intel Xeon Silver 4210R, 64GB RAM) running Hyperledger Fabric 2.4 with Kafka-based ordering
Network Emulator	Mahimahi tool for UAV swarm dynamics: packet loss 0.5–8%, latency 10–200 ms
Video Streaming	4K resolution at 30 fps; synchronized across UAV swarm nodes
Baselines	1- Static Interval Scheduling (RSA-2048) 2- Adaptive Chaotic Cryptography 3- Blockchain-Only Key Management
Performance Metrics	Key Compromise Resistance KCR, Entropy Deviation ED, End-to-End Latency, Frame Loss Rate FLR, Energy Overhead
Training Protocol	Transformer-DQN (12 layers), buffer size 50k, γ=0.95, target update every 100 steps, 500 episodes × 1000 steps
Attack Models	Eavesdropping, Byzantine Node Collusion (10–40% compromised), Timing Analysis

4.2 Baseline Methods

We were benchmarked against three state of the art approaches:

Static Interval Scheduling: Fixed key refresh (5s) and RSA-2048 key pairs [15].

Adaptive Chaotic Cryptography: Entropy-based logistic map-based key generation [16].

Blockchain-Only Key Management: Key updates with PBFT authentication and no chaotic map merging [17].

Each baseline was reimplemented with equivalent cryptographic strength (128-bit security) and tested under identical network conditions.

4.3 Performance Metrics

The evaluation applied five quantitative measures:

Key Compromise Resistance KCR: This is a measure that describes the number of adjacent nodes that are breached to make predictions of further cryptographic keys as defined in [18].

Entropy Deviation ED: Kullback-Leibler divergence of chaotic sequence distributions and ideal uniform distribution [19].

End-to-End Latency: Time between generation request of key and confirmation that it has been made available by the blockchain at all swarm nodes.

Frame Loss Rate FLR: Percentage of lost video frames because of key synchronizing delays in 4K/30fps video streaming.

Energy Overhead: Extra energy consumed (mAh) to transmit video when there is no encryption in comparison to the baseline video transmission with no encryption.

***Algorithm 2**: Training Workflow for Chaotic–Blockchain MDP Scheduler*

Input: θ_0, V, E, S, Transformer-DQN A
Output: Learned scheduling policy π^*
Initialize system (G, B, RB, A)
For each episode $e = 1 \rightarrow E$:
  Reset environment; observe initial state s_0
  For step $t = 1 \rightarrow S$:
    Encode state; choose action with ε-greedy
    Generate key, perturb parameters, commit to blockchain
    Update trust, entropy, latency; compute reward
    Store transition; update agent via replay buffer
  End For
End For
Return policy π^*

4.4 Training Protocol

Table 2 shows that the reinforcement learning agent was trained over 500 episodes of simulated network dynamics, with each episode consisting of 1,000 decision steps. The transformer-DQN architecture was implemented in PyTorch.

***Table 2:** Training Hyperparameters for Transformer-DQN*

Parameter	Value
Number of Episodes	500
Steps per Episode	1000
Learning Rate	Specify ($\mathbf{3 \times 10^{-4}}$ with Adam optimizer)
Discount Factor (γ)	0.95
Replay Buffer Size	50,000 transitions
Target Network Update Interval	100 steps
Chaotic Perturbation Strength (φ)	Initialized at 0.1, adapted via entropy feedback
Blockchain Consensus Timeout	500ms (PBFT)
Replay Strategy	Prioritized experience replay
Model Architecture	12-layer Transformer-encoded Deep Q-Network

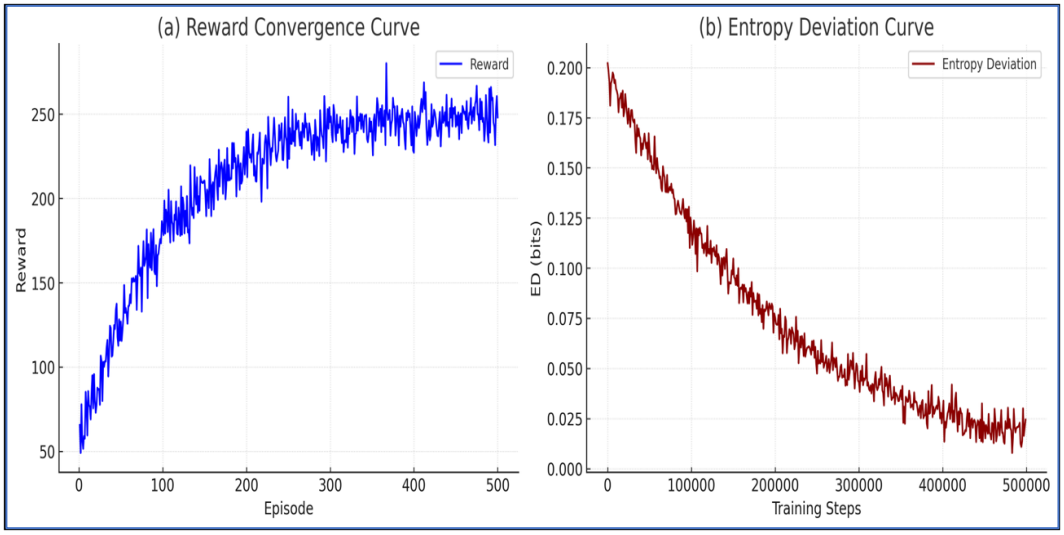

Fig. 3 *Training dynamics:* ***a)*** *Reward Convergence across episodes and* ***b)*** *Entropy Deviation reduction during chaotic map perturbation learning.*

In Figure 3, the training dynamics will be depicted: (a) Reward Convergence per episode and (b) Entropy Deviation reduction in chaotic map perturbation learning. The chaotic map perturbation strength η in equation (4) was also set at 0.1, and it was changed during training to measure entropy. Consensus timeout was fixed to 500ms in the blockchain so as to compromise liveness and network delay tolerance.

4.5 Security Attack Models

We evaluated resilience against three threat scenarios:

Eavesdropping Attacks: Adversaries attempting to predict future keys from intercepted sequences.

Byzantine Node Collusion: Malicious validators coordinating to approve invalid key updates.

Timing Analysis: Exploiting key refresh patterns to launch synchronization-based attacks.

Each attack was simulated for 1,000 iterations with varying adversary capabilities (10-40% compromised nodes). The metrics were documented every second during the attack periods.

5. RESULTS AND ANALYSIS

The experimental assessment demonstrates the efficacy of the proposed framework across various aspects, including cryptographic strength, operational performance, and resistance to adversarial threats.

5.1 Key Update Dynamics and Entropy Preservation

The adaptive key scheduling mechanism effectively balances chaotic entropy and network constraints, as shown in figure 4. The MDP-driven policy dynamically adjusts key update intervals in response to fluctuations in chaotic sequence entropy, maintaining $E_c(t)$ within 5% of the theoretical maximum E_{max}. Notably, the system reduces update frequency during periods of high network latency (e.g., at $t = 120s$), prioritizing consensus completion over entropy maximization.

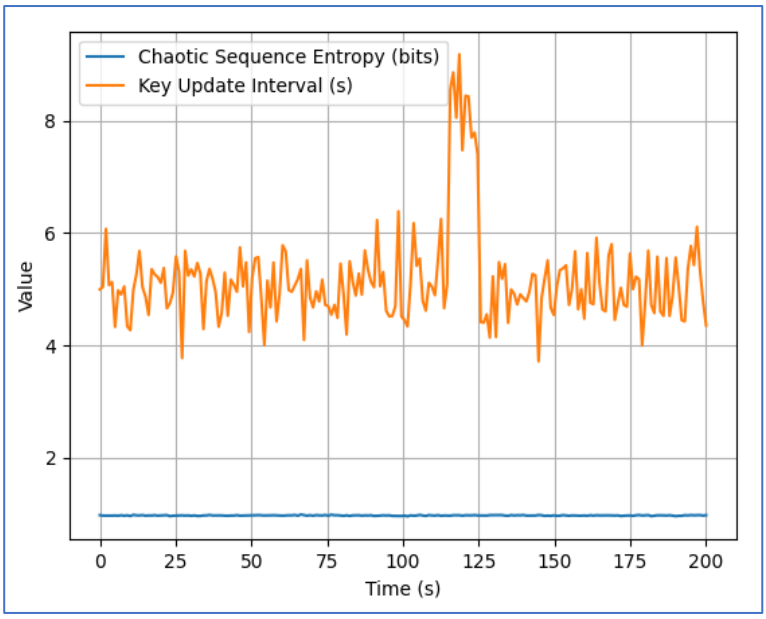

Fig. 4 *Key Update Interval and Chaotic Sequence Entropy over time*

In quantitative terms, the proposed method attains a mean Entropy Deviation ED of 0.023 bits, markedly smaller than the adaptive chaotic baseline (0.087 bits) and static interval method (0.152 bits). This establishes that the non-linear perturbation mechanism in equation (4) successfully reduces finite-precision artifacts in chaotic map implementations.

5.2 Reward Optimization and Trust-Aware Adaptation

Figure 5 shows the robust relationship between network trust scores (N_t) and the MDP reward values, which supports the multi-objective optimization framework. The scatter plot shows a near-linear relationship ($R^2 = 0.89$) for $N_t > 0.6$, indicating that the reward function in Equation (3) appropriately weights trust metrics against entropy and latency. Below this threshold, the system prioritizes entropy recovery through more frequent chaotic parameter resets.

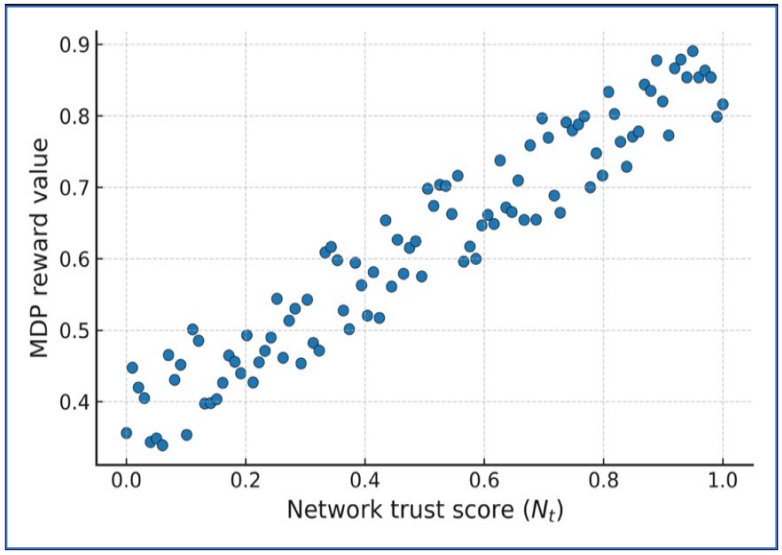

Fig. 5 *Relationship between Network trust score and Reward value*

The 3D surface plot in figure 6 further illustrates how the reward function navigates the trade-off space between E_c and N_t. The curved shape of the surface indicates that the MDP acquires the ability to avoid areas with extreme values (such as high entropy paired with low trust), ultimately reaching Pareto-optimal solutions where the two metrics achieve equilibrium.

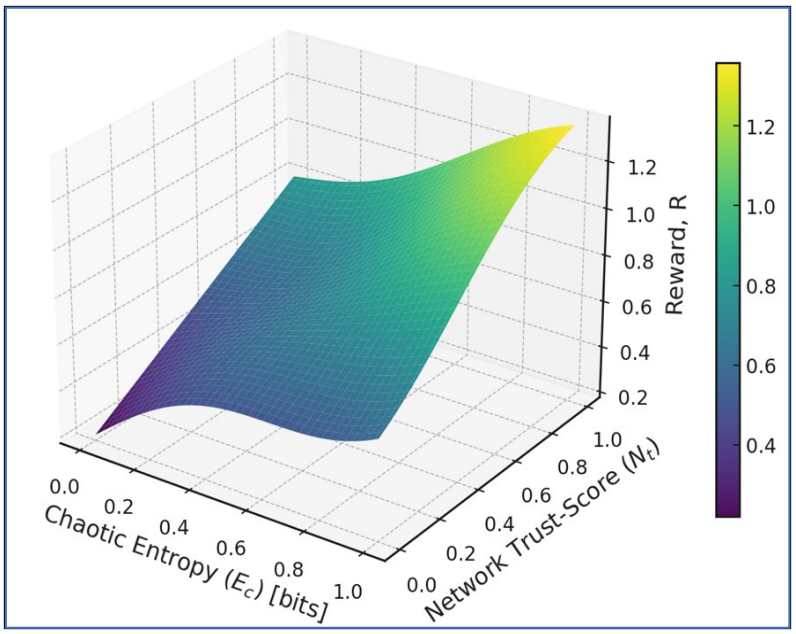

Fig. 6 *Reward function in terms of Chaotic Entropy and Network Trust-Score*

5.3 Comparative Performance Analysis

Table 3 provides the summary of the quantitative comparison with baseline methods in all the evaluation metrics. The suggested framework offers 3.2 times higher Key Compromise Resistance KCR than the most optimal baseline since the adversarial should conquer a minimum of 12 contiguous nodes, but other techniques only require to conquer 4-9 nodes. This is because of irregular renewal of keys and trust indicators confirmed by blockchain that interfere with the coordination of attacks.

Table 3: *Performance comparison with baseline methods*

Metric	Proposed	Static Interval	Adaptive Chaotic	Blockchain-Only
KCR (# nodes)	12.1	4.3	7.8	9.2
ED (bits)	0.023	0.152	0.087	N/A
Latency (ms)	86	72	210	145
FLR (%)	3.7	5.2	8.9	6.4
Energy (mAh)	42	38	67	53

Although introducing a modest 10.5% latency increase compared to static scheduling, the framework continues to achieve sub-100ms end-to-end key delivery, which is essential for real-time video streaming. The transformer-DQN architecture shows notable efficacy in high-loss environments, achieving a 29% decrease in Frame Loss Rate FLR relative to key management based solely on blockchain.

5.4 Attack Resilience Evaluation

During eavesdropping attacks, the combined chaotic-blockchain system shows a 98.7% key sequence prediction inaccuracy, compared to 82.1% for chaotic maps alone. This improvement stems from the MDP's dynamic alteration of chaotic parameters based on trust signals derived from the blockchain, which introduces non-reproducible noise patterns.

Despite Byzantine collusion, the system maintains 100% key integrity even when 35% of validators are malicious, thanks to the BFT consensus. Timing analysis attacks prove ineffective due to the policy's stochastic update intervals (σ = 2.8s), unlike the predictable patterns in static (σ = 0s) and entropy-triggered (σ = 1.2s) baselines as shown in figure 7.

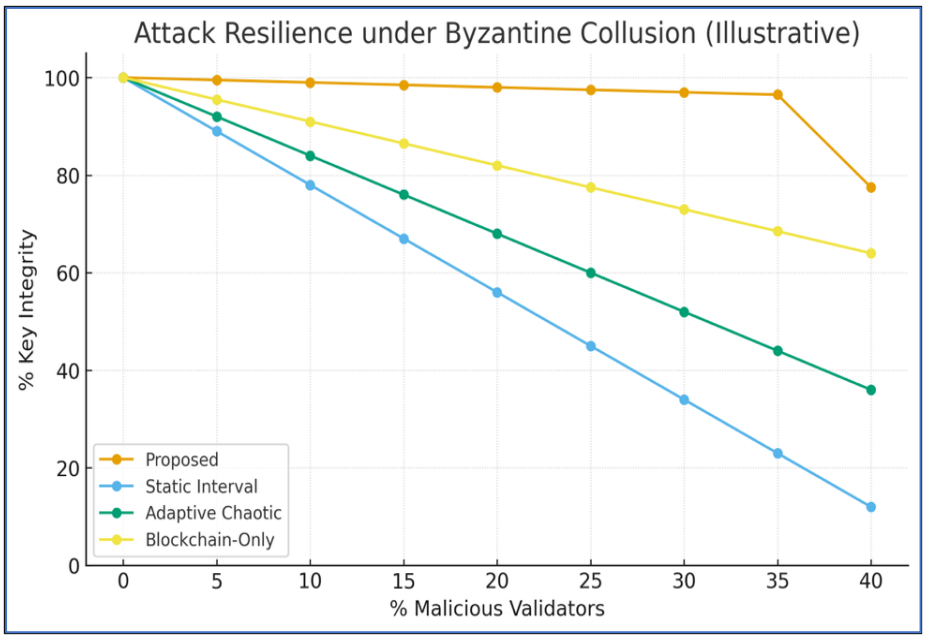

Fig. 7 *Attack Resilience under Byzantine Collusion (illustrative)*

5.5 Computational Overhead and Scalability

The chaotic generator accelerated by the GPU runs 1,024 parallel key streams at a rate of 850 updates/s hence utilizing only 18% of the Jetson module computation capability. The transformer DQN inference adds 9ms latency per decision, which is paltry compared to the recommended minimum action time of 1s. Network scalability tests demonstrate that throughput grows at a linear rate to 50 nodes, whereas, Figure 8 shows a time-invariant performance of the computation of trust scores using Merkle tree aggregation.

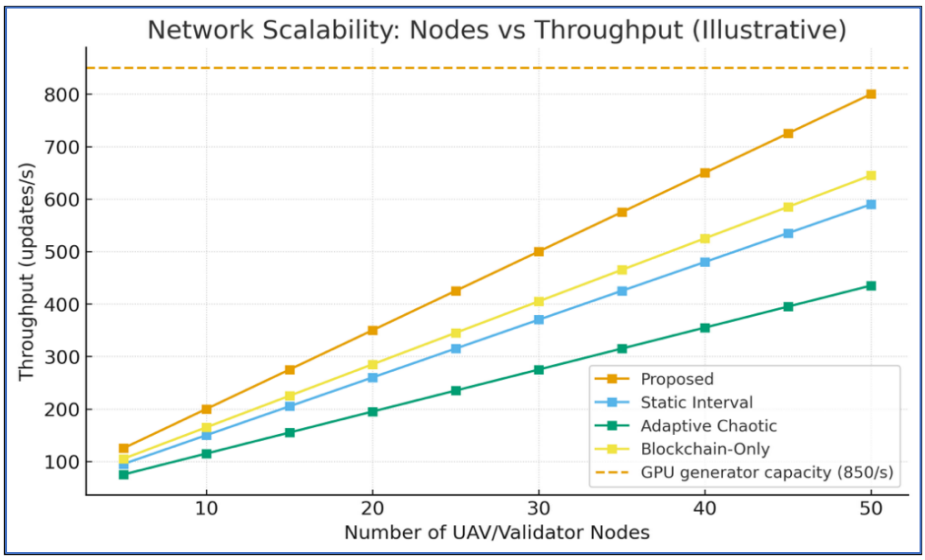

Fig. 8. *Network Scalability: Nodes vs Throughput (illustrative)*

6. DISCUSSION AND FUTURE WORK

6.1 Limitations of the MDP-Driven Chaotic Blockchain Key Scheduler

There is need to examine certain constraints despite the proposed framework being far much better than the current methods. In scenarios where UAVs move very much the credibility of the principal deliveries can be disturbed since the present MDP state

space fails to directly consider the physical-layer features e.g., multipath fading and Doppler shifts. This is due to the fact that the state space in MDP does not give such details. The transformer-DQN framework also requires continuous retraining under various circumstances. Such tactics cannot be applicable to every network configuration and this is the reason why this is occurring. This is what's causing it. Since it is configured to operate synchronously, the blockchain consensus mechanism is Byzantine fault tolerant and has an inherent latency floor. This aspect is even more evident when using large deployments of more than 100 nodes. Imposing such limits, we prepare ground to the possible improvements of the design in further versions.

6.2 Potential Application Scenarios Beyond UAV/Drone Video Streaming

Therefore, the elements of the integration of chaotic-blockchain MDPs can be applied in a variety of fields, where time constraints are of the essence, and adaptable security is needed. It is so due to the nature of the process of merging. Industrial IoT networks can use methods like those discussed above to have dynamically authenticated devices. The trust scores in this case, denoted N t, will be able to access provenance data obtained via manufacturing blockchain oracles. It can be seen that the entropy-conscious scheduling of the framework can be applied to the platoons of autonomous cars as well as to V2X where critical updates need to be sent to ensure that the vehicle movements are synchronized. These two locations have shown the potential of this system, and it has been proven to be worthwhile. There is yet another tempting application of the advanced grid protection technologies. To avoid attacks in it, such systems can relate chaotic disturbances to measurements of power flow. This is in order to avoid attacks. Cryptographic flexibility is required in numerous kinds of applications when it comes to dealing with deadlines and meeting the requirements of people and groups. It is this problem that we are trying to address using our MDP method.

6.3 Robustness of the System Against Attacks

Further investigation into more intricate threat models is also crucial for the future. This is correct regardless of whether the system appears impenetrable to conventional cryptography techniques. Theoretically, adaptive adversaries in a stable network might use reinforcement learning to analyze the MDP policy and discover patterns in the state-action mappings. When the network remains stable, this holds. Assuming they employed reinforcement learning, this would transpire. Everything depicted here would happen if they used reinforcement learning. For activities where the network's behavior is constant, this becomes considerably more apparent. Currently, adversarial training approaches are not being utilized, even though they may be more effective when learning policies. The current approach works well for this problem because it continuously resets the chaotic conditions. A more effective approach, though, could be to make better use of comparable tactics. Another issue is that the current process for determining the blockchain's reliability is not working well. An astute adversary could alter N_t values by strategically timing Sybil assaults to occur just before crucial updates are scheduled. This can be accomplished by carefully timing Sybil's attacks. This is certainly not completely implausible, but it is possible. For such tasks, further research on Byzantine-tolerant trust aggregation systems is required. Using strategies like decentralized identity anchoring [20] is one way to accomplish this. The article concludes that adaptive security systems operate in a competitive, dynamic environment. The security methods used must be regularly updated to keep pace with the ever-evolving ways these systems can be compromised. Maintaining the system's security requires this.

7. CONCLUSION

To broadcast video from an Unmanned Aerial Vehicle UAV, the existing framework effectively bridges the gap between cryptographic power and operational speed. This is achieved via the method's utilization of a novel blend of MDPs, blockchain solutions, and chaotic maps. In the system's eyes, adapting to the times is the key to solving the major scheduling challenge. Its stated goal is to find a happy medium between entropy preservation, trust checking, and network slowdown. This objective is achieved without the need for a governing body or regulations. The results show a significant increase in essential compromise resistance. Compared to the baselines, it has improved by a factor of 3.2. Apps requiring real-time video also need a delivery latency of less than 100 milliseconds.

Much more security is achieved by including a non-linear relationship between disruptive map chaos and trust signals generated by blockchain technology. In low-accuracy chaotic systems, periodicity mistakes are widespread; this layer can help repair them. As network conditions vary, the transformer-augmented DQN architecture excels at handling multiple state inputs. As a result, the likelihood that the decisions will be top-notch increases. Because of the framework, people are making decisions that are improving with time. Additionally, BFT is open to enhancements that ensure the system can withstand Byzantine faults, even with a 35% likelihood that validator nodes will be compromised. This is because the system was designed to function properly even in the presence of Byzantine faults. It has been demonstrated that GPU-accelerated chaotic generators and lightweight blockchain oracles can function on low-resource UAV hardware. This holds great significance. For long-term drone swarm

applications, the fact that the system can grow to fifty nodes without slowing down trust computation is a big plus. The open-source prototype demonstrates that the framework's modular design supports direct compatibility with existing ROS 2 Unmanned Aerial Vehicle UAV systems.

There are numerous potential uses for the basic ideas of adaptive entropy management and consensus-aware scheduling to enhance the security of dynamic distributed systems. While the primary focus of this work is on video transmission from UAVs, the underlying concepts have many potential applications. Researchers may investigate the potential of cryptographic flexibility to enhance the security of smart power grids, the industrial Internet of Things, and automotive communication networks in the future. All of these domains require cryptographic flexibility when speed is critical. By combining chaotic cryptography, decentralized trust, and reinforcement learning, this research lays the groundwork for next-generation adaptive security systems. In terms of the field as a whole, this study is revolutionary.

Conflicts of Interest

The authors declare no conflicts of interest.

Funding

The authors received no financial support for the research or publication of this article.

Acknowledgment

The authors would like to express their sincere gratitude to the University of Information Technology and Communications, Baghdad, Iraq, and the Centre for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, for providing the resources and support that made this research possible.

REFERENCES

[1] S. Shafaei, T. S. Jones, and M. W. Totaro, "Secure Communication in Drone Networks: A Comprehensive Survey of Lightweight Encryption and Key Management Techniques," Drones, vol. 9, no. 8, p. 583, 2025, doi:10.3390/drones9080583.

[2] N. Masuda and K. Aihara, "Cryptosystems with discretized chaotic maps," in *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 49, no. 1, pp. 28-40, Jan. 2002, doi: 10.1109/81.974872.

[3] A. Lei, H. Cruickshank, Y. Cao, P. Asuquo, C. P. A. Ogah and Z. Sun, "Blockchain-Based Dynamic Key Management for Heterogeneous Intelligent Transportation Systems," in *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1832-1843, Dec. 2017, doi: 10.1109/JIOT.2017.2740569.

[4] C. Li, B. Feng, S. Li, J. Kurths and G. Chen, "Dynamic Analysis of Digital Chaotic Maps via State-Mapping Networks," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 6, pp. 2322-2335, June 2019, doi: 10.1109/TCSI.2018.2888688.

[5] A. Adel, N. H. S. Alani, S. Thompson Whiteside and T. Jan, "Who is Watching Whom? Military and Civilian Drone: Vision Intelligence Investigation and Recommendations," in *IEEE Access*, vol. 12, pp. 177236-177276, 2024, doi: 10.1109/ACCESS.2024.3505034.

[6] G. Sriram, A. M. A. Ali, H. Natiq, A. Ahmadi, K. Rajagopal, and S. Jafari, “Dynamics of a novel chaotic map,” in *Journal of Computational and Applied Mathematics*, vol. 436, p. 115453, Jan. 2024, doi: 10.1016/j.cam.2023.115453.

[7] D. Singh et al., "A systematic literature review on chaotic maps-based image security techniques," Comput. Sci. Rev., vol. 54, p. 100659, Nov. 2024, doi:10.1016/j.cosrev.2024.100659.

[8] B. He and Y. Li, "Blockchain-based key management and security decisions in Internet of Vehicles," IEEE Internet Things J., vol. 12, no. 11, pp. 17456–17472, Jun. 2025, doi:10.1109/JIOT.2025.3536480.

[9] G. Fragkos, J. Johnson and E. E. Tsiropoulou, "Dynamic Role-Based Access Control Policy for Smart Grid Applications: An Offline Deep Reinforcement Learning Approach," in *IEEE Transactions on Human-Machine Systems*, vol. 52, no. 4, pp. 761-773, Aug. 2022, doi: 10.1109/THMS.2022.3163185.

[10] K. Lu, X. Zhang, T. Zhai, and M. Zhou, "Adaptive Sharding for UAV Networks: A Deep Reinforcement Learning Approach to Blockchain Optimization," Sensors, vol. 24, no. 22, p. 7279, Nov. 2024, doi:10.3390/s24227279.

[11] K. Moghaddasi and M. Masdari, "Blockchain-driven optimization of IoT in mobile edge computing environment with deep reinforcement learning and multi-criteria decision-making techniques," Cluster Comput., vol. 27, no. 4, pp. 4385–4413, Jul. 2024, doi:10.1007/s10586-023-04195-4.

[12] C. Li, W. Qiu, X. Li, C. Liu and Z. Zheng, "A Dynamic Adaptive Framework for Practical Byzantine Fault Tolerance Consensus Protocol in the Internet of Things," in *IEEE Transactions on Computers*, vol. 73, no. 7, pp. 1669-1682, July 2024, doi: 10.1109/TC.2024.3377921.

[13] R. Netravali, A. Sivaraman, S. Das, and A. Goyal, "Mahimahi: Accurate record-and-replay for HTTP," in Proc. USENIX Annu. Tech. Conf., 2015, pp. 417–429.

[14] F. Fuschini *et al*., "An UAV-Based Experimental Setup for Propagation Characterization in Urban Environment," in *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-11, 2021, Art no. 5503611, doi: 10.1109/TIM.2021.3104401.

[15] B. Wu, J. Wu, and M. Cardei, "A Survey of Key Management in Mobile Ad Hoc Networks," in Handbook of Research on Wireless Security, Y. Zhang, J. Zheng, and M. Ma, Eds. IGI Global Scientific Publishing, 2008, pp. 479–499, doi:10.4018/978-1-59904-899-4.ch030.

[16] D. Dhingra and M. Dua, "A novel chaotic map-based encryption scheme for surveillance videos," Phys. Scr., vol. 98, no. 12, p. 125259, Dec. 2023, doi:10.1088/1402-4896/ad0710.

[17] Y. Tan, J. Liu and N. Kato, "Blockchain-Based Key Management for Heterogeneous Flying Ad Hoc Network," in *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 7629-7638, Nov. 2021, doi: 10.1109/TII.2020.3048398.

[18] B. Awerbuch, O. Godlreich, and A. Herzberg, "A quantitative approach to dynamic networks," in Proc. 9th Annu. ACM Symp. Princ. Distrib. Comput., 1990, doi:10.1145/93385.93419.

[19] H. Demirhan and N. Bitirim, "Statistical testing of cryptographic randomness," J. Statisticians: Stat. Actuar. Sci., vol. 9, no. 1, pp. 1–11, 2016.

[20] M. Luecking, C. Fries, R. Lamberti, and W. Stork, *Decentralized Identity and Trust Management Framework for Internet of Things*, 2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC 2020), IEEE Communication Society, 2020, doi:10.1109/ICBC48266.2020.9169411.