A Robust HECC-based Authentication and Key Agreement for UAV-enabled SAR Networks

Ruiyang Huang, *Student Member, IEEE*, Ning Gao, *Member, IEEE*, Qiang Ni, *Senior Member, IEEE*, and Shi Jin, *Fellow, IEEE*

***Abstract*—Unmanned aerial vehicles (UAVs) have become an indispensable tool in search-and-rescue (SAR) missions, playing a crucial role in locating victims and delivering essential supplies in disaster-stricken areas. However, due to communication over open channels, UAVs are exposed to significant security threats, including unauthorized data access, data tampering, information leakage, and jamming attacks, which can disrupt or even compromise the integrity of SAR missions. The authentication and key agreement (AKA) technologies are designed to try to allow that only verified legal entities can access the data securely with a negotiated session key. Despite the advantages of AKA, existing UAV-specific AKA solutions are unable to resist specific attacks that threaten the security of authentication and session keys and are plagued by the unsatisfactory performance given the limited resources of UAVs. To address these issues, based on hyperelliptic curve cryptography (HECC), we propose a robust AKA for the UAV-enabled SAR system and preserve the desired performance as well. With detailed security analysis, we demonstrate the robustness of our scheme under the eCK Adversary Model. Furthermore, the performance analysis shows the efficiency and utility of our design in terms of computational overhead and communication cost, in which it indicates that our scheme can be applied to the UAV-enabled SAR system.**

***Index Terms*—Authentication and key agreement (AKA), eCK adversary model, hyper-elliptic curve cryptography (HECC), unmanned aerial vehicle (UAV)**

I. Introduction

DRIVEN by the rapid development of intelligent systems, advanced communication platforms, and sophisticated sensing technologies, unmanned aerial vehicles (UAVs) have become an increasingly vital tool in urban search-and-rescue (SAR) [1], offering new opportunities to enhance the rescue efficiency and capabilities of decision-making [2], [3]. As illustrated in Fig. 1, in the chaotic aftermath of a disaster, UAVs can be rapidly deployed to traverse inaccessible terrain.

Manuscript received 18 March 2025; revised 16 December 2025; accepted 23 January 2026. Date of publication , 2026; date of current version , 2026. This work was supported in part by the National Science Foundation of China (NSFC) under Grant 62371131, in part by the National Key Research and Development Program of China under Grant 2024YFE0200700, in part by the Program of Zhishan Young Scholar of Southeast University under Grant 2242024RCB0030, and in part by the National Undergraduate Training Programs for Innovation under Grant 202510286159 and 202510286056. The associate editor coordinating the review of this article and approving it for publication was .*(Corresponding author: Ning Gao)*

R. Huang and N. Gao are with the School of Cyber Science and Engineering, Southeast University, Nanjing 210096, China (e-mail: ryhuang_572@seu.edu.cn; ninggao@seu.edu.cn).

Q. Ni is with the School of Computing and Communications, Lancaster University, Lancaster LA1 4WA, U.K. (e-mail: q.ni@lancaster.ac.uk).

S. Jin is with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China (e-mail: jinshi@seu.edu.cn).

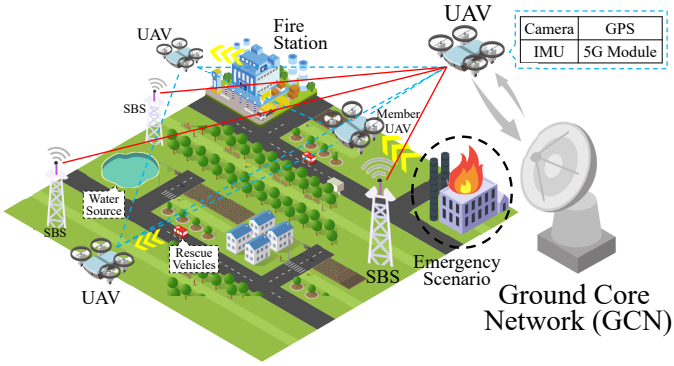

Fig. 1. UAV-enabled SAR network in the smart city scenario.

Equipped with advanced sensors including high-definition and thermal imaging cameras, they capture real-time imagery, identifying the locations of trapped individuals and assessing structural damage from an aerial vantage point [4]. These critical images and videos are relayed back to a ground core network (GCN), which then coordinates nearby resources, such as the internet of vehicles (IoV), to execute time-critical rescue missions. This synergy of multimodal integrated sensing and communications (ISAC) dramatically shortens search times and enhances the precision of rescue operations, directly contributing to saving lives [5], [6].

However, the effectiveness of such a UAV-enabled SAR network is dependent on its communication infrastructure. In disaster scenarios, terrestrial communication infrastructure is often compromised or completely destroyed. This highlights the necessity for decentralized and spontaneous networks where UAVs can communicate directly with each other and with mobile ground units, forming a resilient and self-organizing network. While these open wireless channels provide essential flexibility and rapid deployment, they also introduce profound security risks [7]. The broadcast nature of wireless signals exposes the entire network to a range of cyber-attacks, including identity spoofing, data interception, and man-in-the-middle (MITM) attacks [8]–[13]. For example, a 2023 vulnerability in DJI drones allowed attackers to perform MITM attacks to intercept unencrypted video streams and flight logs, highlighting the risk of critical data breaches and mission sabotage [14]. In an SAR context, such an attack could lead to catastrophic outcomes, from misdirecting rescue teams to exposing sensitive victim information.

A. Motivation and Contribution

To counter these threats, an Authentication and Key Agreement (AKA) protocol provides the definitive solution. An AKA mechanism ensures that a UAV only communicates with legitimate entities through mutual authentication, and it establishes a secure, encrypted channel by negotiating a shared session key. While numerous AKA schemes have been proposed for UAVs [15]–[18], those designed schemes for general-purpose use often falter in the extreme conditions of SAR missions because their high computational cost induces unacceptable latency for resource-constrained UAVs. More critically, they often fail to protect against newer, more sophisticated threats. For instance, node capture attacks, where a physically captured drone's long-term secrets are extracted to compromise the entire network, and ephemeral secret leakage (ESL) attacks, where the leakage of temporary secrets compromises session key security, pose significant risks.

Motivated by these challenges, this study proposes a novel, lightweight, and robust identity authentication scheme for UAV-enabled SAR networks based on the efficiency of hyperelliptic curve cryptography (HECC). Our work directly addresses the motivational questions outlined above and makes the following primary contributions:

1) We first conduct a rigorous security analysis of a recent and relevant scheme proposed by Muhammad *et al.* [17]. We demonstrate that their scheme, despite its merits, remains vulnerable to critical threats, including ESL attacks and Type-I node capture attacks [19]. This analysis reveals a crucial security gap and establishes the need for a more resilient scheme.
2) To address these identified loopholes, we design a robust and efficient AKA scheme tailored for UAV-SAR systems. Our proposed solution facilitates secure mutual authentication between UAVs and negotiates a shared session key to protect the confidentiality and integrity of all subsequent communications. We provide a formal security proof under the extended Canetti-Krawczyk (eCK) adversary model to demonstrate that the session key achieves semantic security. Furthermore, a comprehensive heuristic analysis confirms the scheme's resilience against a wide array of known attacks.
3) We validate the scheme's practicality through a detailed comparative analysis against six baseline schemes encompassing both theoretical costs and simulation-based performance. The results show that our proposed scheme achieves the lowest communication cost and the second-lowest computational cost, offering a superior security guarantee compared to the scheme with the absolute lowest computational cost.

B. Organization of the Paper

The remainder of this paper is structured as follows. Section II is the related works. Section III presents the system architecture. Section IV provides a detailed cryptanalysis of the scheme by Muhammad *et al.* [17] to demonstrate its vulnerabilities. Our proposed scheme is then detailed in Section V, with a comprehensive security analysis in Section VI. The performances are evaluated against six baselines in Section VII. Finally, Section VIII concludes the study.

II. Related Works

In recent years, with the rapid advancement of UAV technology, numerous AKAs have been proposed to mitigate the risks associated with unauthorized access, data breaches, and malicious attacks [20]. These schemes primarily address the security challenges in identity authentication and session key establishment.

Specifically, Pu *et al.* introduced the PMAPD2D [21], which facilitates mutual authentication and session key establishment between UAVs. However, this scheme lacks comprehensive consideration for the entire communication lifecycle, leaving potential vulnerabilities during extended operations. Zhang *et al.* proposed a lightweight AKA scheme [22] that reduces computational overhead, but its security analysis is incomplete, leaving uncertainties about its robustness. Similarly, Gope *et al.* designed a privacy-preserving AKA scheme [18] utilizing edge computing, but the high computational complexity of the solution poses challenges for implementation in resource-constrained UAV environments.

Following, researchers have designed AKA schemes for specific application scenarios. Jan *et al.* developed an AKA scheme [23] for civilian UAVs; however, it cannot adequately address privacy protection, potentially exposing sensitive operational data. Khan *et al.* proposed a certificate-based access control and AKA scheme [17], but its scalability is limited, making it less suitable for large-scale UAV networks. Zhang *et al.* presented an anonymous AKA scheme [24] for intelligent UAVs, but its reliability and practical deployment remain unverified.

Some works have explored PUF-based lightweight AKA schemes to enhance efficiency. Wang *et al.* introduced a multi-factor and PUF-based AKA scheme [25] that significantly improved authentication speed. Ever *et al.* proposed a lightweight AKA scheme [26] for mobile sink nodes, but its security analysis requires further improvement to establish its robustness against advanced attacks.

Furthermore, blockchain-based cross-domain AKA scheme has garnered attention for their potential to secure complex UAV networks. Feng *et al.* proposed a blockchain-based cross-domain AKA scheme [27], enabling secure registration, authentication, and auditing of UAVs. However, its real-time performance and efficiency remain areas for improvement. Tanveer *et al.* introduced a three-factor AKA scheme [28] to enhance security, but its high computational overhead makes it unsuitable for resource-constrained UAV systems.

Overall, these AKA schemes provide effective security measures for UAV communication but exhibit limitations. Specifically, most schemes focus on a single security objective, such as identity authentication or session key establishment, without addressing the entire communication cycle comprehensively. Additionally, the high computational complexity and communication cost of some schemes hinder their applicability in resource-constrained environments.

TABLE I
DESCRIPTIONS OF SYMBOLS IN OUR SCHEME

Symbols	Descriptions
$\mathcal{CA}$	trusted certifier's authority
$\mathcal{DRN}_i$	the i^{th} drone
ID_i	unique identity of $\mathcal{DRN}_i$
$\mathcal{C}r_i$	certificate of $\mathcal{DRN}_i$
(a_i, A_i)	private and public master key of $\mathcal{DRN}_i$
$(d_{\mathcal{CA}}, w_{\mathcal{CA}})$	private and public key of the certifier's authority
D	the generator of the hyper-elliptic curve
$k{\cdot}D$	scalar multiplication on the hyper-elliptic curve
$h(\cdot)$	the secure hash function
f_i	random salt of $\mathcal{DRN}_i$
(ω_i, Ω_i)	$\mathcal{DRN}_i$'s private and public ephemeral keys
ω_i^*	$\mathcal{DRN}_i$'s temporary salt of the session
Λ_i	$\mathcal{DRN}_i$'s signature for the session
Ψ_i	the i^{th} message in the public channel
$\mathcal{SK}_{uv}$	shared session keybetween $\mathcal{DRN}_u$ and $\mathcal{DRN}_v$
$\mathcal{E}_\kappa(\cdot)$	encryption operation with the key κ
$\mathcal{D}_\kappa(\cdot)$	decryption operation with the key κ
ΔT	the timeout duration for retransmission
C_{max}	the maximum number of retries
δ	a small, configurable time buffer
$X \| Y$	the concatenate operation of string X and Y
$\longrightarrow$	private channels in the secure environment
$\Longrightarrow$	public channels in the open environment

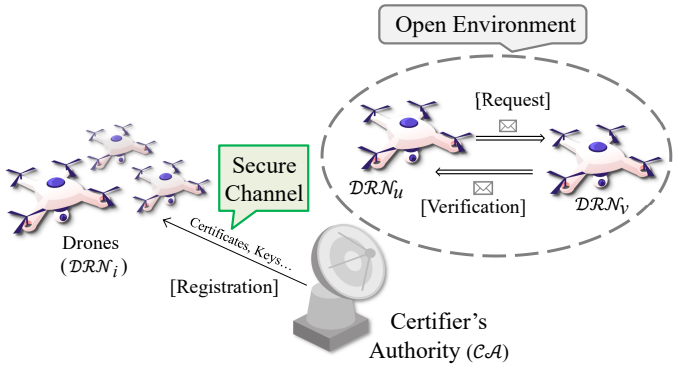

Fig. 2. Network model of the proposed scheme.

III. PRELIMINARIES

A. Network Model and System Architecture

The system model for our proposed UAV-enabled SAR network is composed of two primary entities: a trusted certifier's authority ($\mathcal{CA}$) and a dynamic swarm of drone nodes ($\mathcal{DRN}$s), as depicted in Fig. 1. The $\mathcal{CA}$ acts as an offline, centralized root of trust, responsible for generating and securely provisioning cryptographic credentials including identities, private keys, and certificates to all legitimate drones during a pre-deployment registration phase. Once deployed, the drone nodes operate in a decentralized manner, forming a flying ad-hoc network (FANET) to collaboratively execute their missions. For the sake of clarity, the symbols and their descriptions used in our system are illustrated in Table I.

Designed for SAR scenarios, our architecture consists of a homogeneous swarm of N UAVs that form a self-organizing, multi-hop mesh. Each drone possesses sufficient computational resources to execute our HECC-based AKA protocol, with a communication stack built on the *IEEE 802.11b* standard in ad-hoc mode to ensure low-latency message exchange. To manage the dynamic FANET topology, the optimized link state routing protocol (OLSR) is employed at the network layer for efficient packet forwarding, while authentication messages are transmitted over a designated UDP port using unique IP addresses.

In a typical SAR mission, the drone swarm works in concert to achieve a common objective, such as mapping a disaster area or locating survivors. A drone, acting as an initiator (e.g., $\mathcal{DRN}_u$), may need to establish a secure communication channel with another drone (e.g., $\mathcal{DRN}_v$) to exchange tactical information, such as sensor data or flight path adjustments. To do this, $\mathcal{DRN}_u$ initiates our proposed AKA protocol. It generates and transmits the first authentication message, Ψ_1, containing its credentials and a fresh nonce, to $\mathcal{DRN}_v$. Upon receiving Ψ_1, $\mathcal{DRN}_v$ validates the message using the public parameters provisioned by the $\mathcal{CA}$, performs its own cryptographic calculations, and sends back a response message, Ψ_2. Finally, upon receiving a valid Ψ_2, $\mathcal{DRN}_u$ completes the authentication process. At this point, both drones have mutually authenticated each other and have independently computed an identical, shared session key. This lightweight two-way handshake enables the establishment of a secure and end-to-end encrypted channel.

B. Threat Model

In the design of the authentication protocols, the extended Canetti-Krawczyk (eCK) model proposed by LaMacchia *et al.* [29] is used for evaluating the security of cryptographic protocols, which aims to take into account possible attack methods and attacker's capabilities more comprehensively. Specifically, it has been proved in reference [30] that the adversary's greater ability than the Dolev-Yao (DY) model. Therefore, our proposed scheme based on the eCK model will exhibit more robust security compared to the conventional AKA schemes based on the DY model. In this more complete threat model, Attacker $\mathcal{A}$ can be entitled to the six capabilities in Table II [31].

IV. CRYPTANALYSIS OF MUHAMMAD ET AL.'S SCHEME

In this section, we focus on the security of Muhammad *et al.*'s scheme and then demonstrate the two security threats found from the drone access control process in Muhammad *et al.*'s scheme. Given the limited space, in the following, we only show three attack paths, and the description of drone access control process can be found in Muhammad *et al.*'s scheme.

A. No Resistance to ESL Attacks

In the AKA scheme, an ESL attack, refers to a scenario in which an adversary gains access to one of the temporary private key between the two communication parties. Once the adversary has obtained this type of key, the session key $\mathcal{K}_{uv}$ in Muhammad *et al.*'s scheme can be revealed by the adversary under the following circumstances.

Attack target: get the session key $\mathcal{K}_{uv}$.

TABLE II
THREAT MODEL: DESCRIPTION OF ATTACKERS' CAPACITIES

C_*	Description of Attackers' Capacities
C_1	$\mathcal{A}$ can fully control the public channel and then intercept, modify, insert, and delete any messages transmitted in the open channel
C_2	$\mathcal{A}$ can masquerade as a legal UAV to communicate in the open channel
C_3	$\mathcal{A}$ can obtain the long-term private key of the certifier's authority $\mathcal{CA}$
C_4	$\mathcal{A}$ can acquire previous session keys between the two $\mathcal{DRN}$s
C_5	$\mathcal{A}$ can learn one side of the $\mathcal{DRN}$s' temporary private key in a session when considering the system's eventual failure
C_6	$\mathcal{A}$ can break some UAV nodes, extract the stored sensitive data, and even control the broken node to join the next communication interaction

Attacker's capability:

- Obtain $\mathcal{DRN}_u$'s temporary private key ε_u or $\mathcal{DRN}_v$'s temporary private key ε_v;
- Eavesdrop these messages from the public channel: $\Psi_1 = (NON_{eu}, \Omega_u, \Lambda_u, b_u, \mathcal{X}_u, \mathcal{C}r_u)$, $\Psi_2 = (NON_{euv}, \Omega_v, \Lambda_v, b_v, \mathcal{X}_v, \mathcal{C}r_v, \mathcal{VK}_{uv}, ID_v)$, $\Psi_3 = (NON^*_{euv}, \mathcal{VK}^*_{uv})$.

Attack path:

1) Obtain $\mathcal{DRN}_v$'s temporary private key ε_v;
2) Base on the relation relevant to Λ_v, and compute
$$a_v = \frac{\Lambda_v - \mathcal{C}r_v}{h(\mathcal{X}_v||\Omega_v||\mathcal{C}r_v||b_v||NON_{euv})} - \varepsilon_v;$$
3) Use a_v to decrypt $(N_u, ID_u) = \mathcal{D}_{a_v}(NON_{eu})$ and get ID_u;
4) Use a_v to compute $\mu_{uv} = a_v{\cdot}b_u = a_v a_u{\cdot}D$;
5) Use ε_v to compute $\gamma_{uv} = \varepsilon_v{\cdot}\Omega_u = \varepsilon_v\varepsilon_u{\cdot}D$;
6) Compute $\mathcal{K}_{uv} = h(ID_u||NON_{euv}||\gamma_{uv}||\mu_{uv}||ID_v)$.

Time complexity: $O(2T_h + 2T_p + T_{\mathcal{D}})$. Here, T_h, T_p, and $T_{\mathcal{D}}$ denote the time for hash function, additive group's multiplicative operation, and decryption, respectively.

B. No Resistance to Node Capture Attacks

The current research by Wang [32] on node capture attacks describes the Type-I attack, in which the adversary compromises the UAV node's secret value a_u and a_v and eavesdrops on messages from the open channel. This allows the adversary to leak the session key $\mathcal{K}_{uv}$ of the two communication parties. Furthermore, the scheme proposed by Muhammad *et al.* is susceptible to a Type-I node capture attack.

Attack target: Get the session key $\mathcal{K}_{uv}$.

Attacker's capability:

- Capture and restore $\mathcal{DRN}_u$'s memory $(ID_u, \mathcal{C}r_u, \Lambda_u, a_u, \mathcal{X}_u)$, or $\mathcal{DRN}_v$'s memory;
- Eavesdrop Ψ_1, Ψ_2, Ψ_3 from the public channel.

Attack path 1:

1) Obtain $\mathcal{DRN}_v$'s temporary private key a_v;
2) Base on the relation relevant to Λ_v, and compute
$$\varepsilon_v = \frac{\Lambda_v - \mathcal{C}r_v}{h(\mathcal{X}_v||\Omega_v||\mathcal{C}r_v||b_v||NON_{euv})} - a_v;$$
3) Use a_v to decrypt $(N_u, ID_u) = \mathcal{D}_{a_v}(NON_{eu})$ and get ID_u;
4) Use a_v to compute $\mu_{uv} = a_v{\cdot}b_u = a_v a_u{\cdot}D$;
5) Use ε_v to compute $\gamma_{uv} = \varepsilon_v{\cdot}\Omega_u = \varepsilon_v\varepsilon_u{\cdot}D$;
6) Compute $\mathcal{K}_{uv} = h(ID_u||NON_{euv}||\gamma_{uv}||\mu_{uv}||ID_v)$.

Attack path 2:

1) Obtain $\mathcal{DRN}_u$'s temporary private key a_u;
2) Base on the relation relevant to Λ_u, and compute
$$\varepsilon_u = \frac{\Lambda_u - \mathcal{C}r_u}{h(\mathcal{X}_u||\Omega_u||\mathcal{C}r_u||b_u||NON_{eu})} - a_u;$$
3) Use a_u to compute $\mu_{uv} = a_u{\cdot}b_v = a_u a_v{\cdot}D$;
4) Use ε_u to compute $\gamma^*_{uv} = \varepsilon_u{\cdot}\Omega_v = \varepsilon_u\varepsilon_v{\cdot}D$;
5) Compute $\mathcal{K}^*_{uv} = h(ID_u||NON_{euv}||\gamma^*_{uv}||\mu^*_{uv}||ID_v)$;
6) Compute $\mathcal{VK}^*_{uv} = h(NON_{euv}||\mathcal{K}^*_{uv})$, and check if $\mathcal{VK}_{uv} \stackrel{?}{=} \mathcal{VK}^*_{uv}$.

Time complexity: $O(2T_h + 2T_p + T_{\mathcal{D}})$ and $O(3T_h + 2T_p)$.

V. OUR PROPOSED SCHEME

This section details the architecture of our proposed scheme, which is built upon the security primitives of HECC and a one-way hash function. The protocol is systematically presented through its distinct phases: we first describe the initial set-up and UAV registration. We then elaborate on the core UAV access control process, which is enhanced by a timeout and retransmission mechanism to ensure operational reliability. Finally, the procedure for new drone addition is explained. A detailed explanation of each step is provided below.

A. Set-up Phase of Certifier's Authority

The set-up phase is executed by the trustworthy certifier's authority $\mathcal{CA}$. Its main task is to generate private and public keys. Furthermore, a set of public parameters is prepared in this step. To perform such a calculation, the following sequence needs to be followed:

1) Firstly, the trusted certification authority $\mathcal{CA}$ selects $d_{\mathcal{CA}}$ as its private key from the dataset $\{1, 2, 3, \ldots, n-1\}$.
2) It generates the public key as follows:
$$w_{\mathcal{CA}} = d_{\mathcal{CA}}{\cdot}D,$$
where D is the generator of the hyperelliptic curve.
3) Choose the hash function h, which has the ability to avoid collisions and is characterized by irreversibility.
4) Choose the common parameter set as $\eta = (d_{\mathcal{CA}}, D, n, h)$ and publish it.

B. UAV Registration Phase

The drone registration step considers a deployed drone $\mathcal{DRN}_u$, assuming that the authorized body $\mathcal{CA}$ wants to register it offline. The process proceeds in the following order:

1) For $\mathcal{DRN}_u$, an identity ID_u and a private key a_u are obtained from the dataset $\{1, 2, 3, \ldots, n-1\}$.

$\mathcal{DRN}_u$	Public Channel	$\mathcal{DRN}_v$
Store $(ID_u, \mathcal{C}r_u, A_u, a_u, \mathcal{X}_u)$ into memory		Store $(ID_v, \mathcal{C}r_v, A_v, a_v, \mathcal{X}_v)$ into memory
Phase # 1		
Select ω_u and ω_u^* and calculate:		
$\Omega_u = \omega_u \cdot D$ and $\Omega_u^\circ = (\omega_u + \omega_u^*) \cdot D$	(1) Ψ_1	
Select a fresh nonce N_u	$\Longrightarrow$	***Phase # 2***
Sign for the session:		Decrypt the message Ψ_1 with a_v:
$\Lambda_u = \mathcal{C}r_u + h(ID_u \| \mathcal{C}r_u \| A_u \| \mathcal{X}_u \| N_u)(\omega_u + \omega_u^* + a_u)$		Get $(ID_u, N_u, A_u, \mathcal{C}r_u, \mathcal{X}_u, \Omega_u, \Omega_u^\circ, \Lambda_u)$
Generate Ψ_1:		Compute:
$\Psi_1 = \{\mathcal{E}_{A_v}(ID_u, N_u, A_u, \mathcal{C}r_u), \mathcal{X}_u, \Omega_u, \Omega_u^\circ, \Lambda_u\}$		$\Omega_u^* = \Omega_u^\circ - \Omega_u$
		Check:
		$w_{\mathcal{CA}} + h(ID_u \| \mathcal{X}_u) \cdot \mathcal{X}_u \stackrel{?}{=} \mathcal{C}r_u \cdot D$
	(2) Ψ_2	$\mathcal{C}r_u \cdot D + h(ID_u \| \mathcal{C}r_u \| A_u \| \mathcal{X}_u \| N_u) \cdot (\Omega_u^\circ + A_u) \stackrel{?}{=} \Lambda_u \cdot D$
Phase # 3	$\Longleftarrow$	If it is valid, compute:
Decrypt the message Ψ_2 with a_u:		$\Omega_v^\circ = (\omega_v + \omega_v^*) \cdot D$
Get $(NON_{ev}, \Lambda_v, \mathcal{C}r_v, \mathcal{VK}_{uv}, ID_v, \Omega_v^\circ, A_v, \mathcal{X}_v)$		$\Lambda_v = \mathcal{C}r_v + h(ID_v \| \mathcal{C}r_v \| A_v \| \mathcal{X}_v \| N_v)(\omega_v + \omega_v^* + a_v)$
Decrypt $(N_v, ID_v) = \mathcal{D}_{\omega_u^*}(NON_{ev})$		$\lambda_{uv} = \omega_v \cdot \Omega_u = \omega_v \omega_u \cdot D$ and $\mu_{uv} = a_v \cdot A_u = a_v a_u \cdot D$
Check:		Generate $\mathcal{SK}_{uv}$:
$w_{\mathcal{CA}} + h(ID_v \| \mathcal{X}_v) \cdot \mathcal{X}_v \stackrel{?}{=} \mathcal{C}r_v \cdot D$		$\mathcal{SK}_{uv} = h(\mathcal{C}r_u \| N_u \| \lambda_{uv} \| \mu_{uv} \| N_v \| \mathcal{C}r_v)$
$\mathcal{C}r_v \cdot D + h(ID_v \| \mathcal{C}r_v \| A_v \| \mathcal{X}_v \| N_v) \cdot (\Omega_v^\circ + A_v) \stackrel{?}{=} \Lambda_v \cdot D$		Calculate:
If so, Compute:		$\mathcal{VK}_{uv} = h(N_u \| N_v \| \mathcal{SK}_{uv})$
$\lambda_{uv} = \omega_u \cdot \Omega_v = \omega_u \omega_v \cdot D$ and $\mu_{uv} = a_u \cdot A_v = a_u a_v \cdot D$		Encrypt: $NON_{ev} = \mathcal{E}_{\Omega_u^*}(N_v, ID_v)$
$\mathcal{SK}_{uv}^* = h(\mathcal{C}r_u \| N_u \| \lambda_{uv} \| \mu_{uv} \| N_v \| \mathcal{C}r_v)$		Generate Ψ_2:
$\mathcal{VK}_{uv}^* = h(N_u \| N_v \| \mathcal{SK}_{uv}^*)$		$\Psi_2 = \{\mathcal{E}_{A_u}(NON_{ev}, \Lambda_v, \mathcal{C}r_v, \mathcal{VK}_{uv}), ID_v, \Omega_v^\circ, A_v, \mathcal{X}_v\}$
Check:		
$\mathcal{VK}_{uv}^* \stackrel{?}{=} \mathcal{VK}_{uv}$		
If so, Complete the authentication		
$\mathcal{DRN}_u$ and $\mathcal{DRN}_v$ will share a session key $\mathcal{SK}_{uv}$ after authentication		

Fig. 3. UAV mutual authentication and key agreement process.

2) $\mathcal{CA}$ uses the following relationship to calculate the public A_u:

$$A_u = a_u \cdot D.$$

3) $\mathcal{CA}$ selects f_u from the dataset $\{1, 2, 3, \ldots, n-1\}$ and compute the value of $\mathcal{X}_u$ as:

$$\mathcal{X}_u = (f_u + a_u) \cdot D.$$

4) $\mathcal{CA}$ calculates the certificate for ID_u:

$$\mathcal{C}r_u = d_{\mathcal{CA}} + (f_u + a_u) h(ID_u \| \mathcal{X}_u).$$

5) In the last step, $\mathcal{CA}$ puts the set $(ID_u, \mathcal{C}r_u, A_u, a_u, \mathcal{X}_u)$ into $\mathcal{DRN}_u$ in memory.

C. UAV Access Control Process

Suppose two UAVs $\mathcal{DRN}_u$ and $\mathcal{DRN}_v$ wish to communicate with each other through the AKA mechanism. To carry out this process, there are three stages:

Phase # 1: If $\mathcal{DRN}_u$ wants to establish a shared session key with $\mathcal{DRN}_v$, the operation is performed as follows:

1) First, $\mathcal{DRN}_u$ selects different ω_u and ω_u^* from $\{1, 2, 3, \ldots, n-1\}$ and performs the following calculation:

$$\Omega_u = \omega_u \cdot D,$$

$$\Omega_u^\circ = (\omega_u + \omega_u^*) \cdot D.$$

2) Select a fresh nonce N_u for this session.
3) Yield the signature Λ_u for this session on ω_u and ω_u^* according to the following equation:

$$\Lambda_u = \mathcal{C}r_u + h(ID_u \| \mathcal{C}r_u \| A_u \| \mathcal{X}_u \| N_u)(\omega_u + \omega_u^* + a_u).$$

4) Finally, $\mathcal{DRN}_u$ sends the message $\Psi_1 = \{\mathcal{E}_{A_v}(ID_u, N_u, A_u, \mathcal{C}r_u), \mathcal{X}_u, \Omega_u, \Omega_u^\circ, \Lambda_u\}$ to $\mathcal{DRN}_v$ over the open network, here the public key A_v of $\mathcal{DRN}_v$ is considered to be known to $\mathcal{DRN}_u$.

Phase#2: Then, after receiving the message Ψ_1 from $\mathcal{DRN}_u$, $\mathcal{DRN}_v$ performs the following calculation:

1) First, decrypt the message Ψ_1 using the private key a_v to get $ID_u, N_u, A_u, \mathcal{C}r_u, \mathcal{X}_u, \Omega_u, \Omega_u^*, \Lambda_u$. Meanwhile, use the formula to calculate the temporary public key Ω_u^*:
$$\Omega_u^* = \Omega_u^\circ - \Omega_u.$$
2) Check the certificate of the source drone by applying the following conditions:
$$w_{\mathcal{CA}} + h(ID_u||\mathcal{X}_u)\cdot\mathcal{X}_u \stackrel{?}{=} \mathcal{C}r_u\cdot D.$$
3) Verify the following conditions:
$$\mathcal{C}r_u\cdot D + h(ID_u||\mathcal{C}r_u||A_u||\mathcal{X}_u||N_u)\cdot(\Omega_u^\circ + A_u) \stackrel{?}{=} \Lambda_u\cdot D.$$
4) If the signature is valid, further calculations are performed:
$$\Omega_v^\circ = (\omega_v + \omega_v^*)\cdot D.$$
Here ω_v and ω_v^* belong to the dataset $\{1, 2, 3, \ldots, n-1\}$.
5) Calculate the signature Λ_v using the following relation:
$$\Lambda_v = \mathcal{C}r_v + h(ID_v||\mathcal{C}r_v||A_v||\mathcal{X}_v||N_v)(\omega_v + \omega_v^* + a_v).$$
6) Perform the following calculations to generate λ_{uv} and μ_{uv}:
$$\lambda_{uv} = \omega_v\cdot\Omega_u = \omega_v\omega_u\cdot D,$$
$$\mu_{uv} = a_v\cdot A_u = a_v a_u\cdot D.$$
7) At this point, $\mathcal{SK}_{uv}$ can be calculated based on the relation:
$$\mathcal{SK}_{uv} = h(\mathcal{C}r_u||N_u||\lambda_{uv}||\mu_{uv}||N_v||\mathcal{C}r_v).$$
8) Calculate the value of NON_{ev} and $\mathcal{VK}_{uv}$ based on the relationship:
$$NON_{ev} = \mathcal{E}_{\Omega_u^*}(N_v, ID_v),$$
$$\mathcal{VK}_{uv} = h(N_u||N_v||\mathcal{SK}_{uv}).$$
9) Finally, $\mathcal{DRN}_v$ passes the message Ψ_2 to $\mathcal{DRN}_u$ over the open network where $\Psi_2 = \{\mathcal{E}_{A_u}(NON_{ev}, \Lambda_v, \mathcal{C}r_v, \mathcal{VK}_{uv}), ID_v, \Omega_v^\circ, A_v, \mathcal{X}_v\}$.

Phase#3: In this stage, $\mathcal{DRN}_u$ receives the message Ψ_2 and performs the following calculation:

1) First decrypt the message Ψ_2 using the private key a_u to get $NON_{ev}, \Lambda_v, \mathcal{C}r_v, \mathcal{VK}_{uv}, ID_v, \Omega_v^\circ, A_v, \mathcal{X}_v$. Then proceed to decrypt $(N_v, ID_v) = \mathcal{D}_{\omega_u^*}(NON_{ev})$ to get and check the freshness of N_v.
2) Check the certificate of the drone $\mathcal{DRN}_v$ by computing the following conditions:
$$w_{\mathcal{CA}} + h(ID_v||\mathcal{X}_v)\cdot\mathcal{X}_v \stackrel{?}{=} \mathcal{C}r_v\cdot D.$$
3) Verify the signature by the following conditions:
$$\mathcal{C}r_v\cdot D + h(ID_v||\mathcal{C}r_v||A_v||\mathcal{X}_v||N_v)\cdot(\Omega_v^\circ + A_v) \stackrel{?}{=} \Lambda_v\cdot D.$$
4) If the signature is valid, the following calculations are performed to obtain λ_{uv} and μ_{uv}:
$$\lambda_{uv} = \omega_u\cdot\Omega_v = \omega_u\omega_v\cdot D.$$
$$\mu_{uv} = a_u\cdot A_v = a_u a_v\cdot D.$$
5) At this point, $\mathcal{SK}_{uv}^*$ can be calculated based on the relation:
$$\mathcal{SK}_{uv}^* = h(\mathcal{C}r_u||N_u||\lambda_{uv}||\mu_{uv}||N_v||\mathcal{C}r_v).$$
6) Compute $\mathcal{VK}_{uv}^* = h(N_u||N_v||\mathcal{SK}_{uv}^*)$, while comparing $\mathcal{VK}_{uv}^* \stackrel{?}{=} \mathcal{VK}_{uv}$ to verify the shared session key $\mathcal{SK}_{uv}^*$.

D. Timeout and Retransmission Mechanism

To ensure robustness over lossy wireless open channels, we employ a timeout and retransmission mechanism initiated by $\mathcal{DRN}_u$, which is controlled by two parameters: a maximum number of retries, C_{max}, and a timeout duration ΔT, formally defined as:
$$\Delta T = 2T_{prop}(u, v) + T_{proc,v} + \delta,$$
where:

- $T_{prop}(u, v)$ is the average one-way propagation delay between $\mathcal{DRN}_u$ and $\mathcal{DRN}_v$.
- $T_{proc,v}$ is the maximum computational time required for $\mathcal{DRN}_v$ to perform all calculations in *Phase#2*, from receiving Ψ_1 to sending Ψ_2. This value can be determined through offline benchmarking of the UAV hardware.
- δ is a small, configurable time buffer to account for network jitter and other random delays.

The operational logic for the initiator, $\mathcal{DRN}_u$, unfolds as follows:

1) Initially, $\mathcal{DRN}_u$ sets a retry counter $c \leftarrow 0$, generates the message Ψ_1 with a fresh nonce N_u, transmits it, and starts a timer for the duration ΔT.
2) Upon receiving a valid response Ψ_2 before the timer expires, the timer is canceled and the authentication process continues.
3) If the timer expires, the retry counter is incremented. The process is aborted if $c > C_{\max}$. Otherwise, a new message $\Psi_1^\star$ is generated with a fresh nonce $N_u^\star$ and a re-calculated signature $\Lambda_u^\star$. This message is then sent, and the timer is reset.

The responder, $\mathcal{DRN}_v$, requires no special logic to handle this mechanism. The protocol's inherent requirement to verify the freshness of the nonce N_u ensures that it processes each valid retransmission as a new request and discards any duplicates from previous attempts.

E. New Drone Addition Phase

This phase involves the addition of a new drone $\mathcal{DRN}_{new}$ after the establishment of a UAV network. $\mathcal{CA}$ registers the new drone $\mathcal{DRN}_{new}$ in an offline environment. And the process proceeds stepwise as follows:

1) For $\mathcal{DRN}_{new}$, an identity ID_{new} and a private key a_{new} are obtained from the dataset $\{1, 2, 3, \ldots, n-1\}$.
2) $\mathcal{CA}$ uses the following relationship to calculate the public A_{new}:
$$A_{new} = a_{new}\cdot D.$$

3) $\mathcal{CA}$ selects f_{new} from the dataset $\{1, 2, 3, \ldots, n-1\}$ and compute the value of $\mathcal{X}_{new}$ as:

$$\mathcal{X}_{new} = (f_{new} + a_{new}) \cdot D.$$

4) $\mathcal{CA}$ calculates the certificate for ID_{new}:

$$\mathcal{C}r_{new} = d_{CA} + (f_{new} + a_{new}) h(ID_{new} || \mathcal{X}_{new}).$$

5) In the last step, $\mathcal{CA}$ puts $(ID_{new}, \mathcal{C}r_{new}, A_{new}, a_{new}, \mathcal{X}_{new})$ into $\mathcal{DRN}_{new}$'s memory.

VI. Security Analysis of our Proposed Scheme

In this section, we present a comprehensive analysis of our proposed scheme, encompassing both theoretical and empirical evaluations. Initially, we demonstrate the semantic security of the session key through a provable proof under the real-or-random (ROR) oracle model, thereby establishing the theoretical foundation of our scheme's security. Subsequently, we conduct heuristic analyses to substantiate the robustness of our scheme against a spectrum of known attacks, showcasing its resilience in practical scenarios.

To construct the security proof, we first establish three foundational cryptographic definitions and hardness assumptions upon which our analysis is built. Following this, we present the detailed formal security analysis.

A. The Basic Definitions

Definition 1. *A collision-resistant cryptographic one-way hash function $h : \{0,1\} * \rightarrow \{0,1\}^n$ is a deterministic mathematical function that produces a fixed length output string of n bits against a variable length input string.*

Definition 2. *Hyper-elliptic curve discrete logarithm problem (HECDLP) is a hard problem in which the attacker attempts to extract f from the relation $L = f \cdot D$, where f is the uniformly selected number from $\{1, 2, 3, \ldots, n-1\}$ [33].*

Definition 3. *Hyper-elliptic curve computational Diffie-Hellman problem (HECCDHP) is also a hard problem in which the attacker calculates the value of $xy \cdot D$ when $x \cdot D$ and $y \cdot D$ are known, where x, y are uniformly selected from $\{1, 2, 3, \ldots, n-1\}$ [34].*

B. Formal Security Analysis

1) The detailed basis: In this section and the subsequent formal proof, the term $\mathbb{C}$ is employed to indicate the proposed scheme. In $\mathbb{C}$, there are three entities: two drones ($\mathcal{DRN}_u$ and $\mathcal{DRN}_v$) and a certifier's authority ($\mathcal{CA}$). Then, the simulator $\mathbb{S}$ initializes a hyper-elliptic curve $HE(F_p)$ over a large prime finite field F_p and selects a q-order additive subgroup $\mathbb{G}$ with a generator D, where the length of the large prime q is a security parameter n. Next, the drones $\mathcal{DRN}_u$ and $\mathcal{DRN}_v$ are provided with their own information.

Subsequently, the three entities will instantiate $\mathcal{DRN}_u$ with instance $\prod^u_{\mathcal{DRN}_u}$, $\mathcal{DRN}_v$ with $\prod^v_{\mathcal{DRN}_v}$, and $\mathcal{CA}$ with $\prod^{ca}_{\mathcal{CA}}$ respectively. In the absence of a requirement to distinguish between $\prod^u_{\mathcal{DRN}_u}$, $\prod^v_{\mathcal{DRN}_v}$, and $\prod^{ca}_{\mathcal{CA}}$, any instance may be designated as $\prod^t$. Thereafter, each instance will be simulated as an oracle. In particular, if the input message is valid, incorrect, or null, the oracle's state will manifest as acceptance, rejection, or $\perp$ $(NULL)$, respectively.

Based on the threat model, which describes the eCK adversary's capabilities, we define some indispensable items as a basis for a formal proof that will be used to prove the semantic security.

Accepted state: Upon receipt of the last expected protocol message, the instance $\prod^t$ receives an accepted state. Meanwhile, throughout the communication, the ordered concatenation of all sent and received messages forms the session identifier of $\prod^t$.

Partnering: Two instances $\prod^{t_1}, \prod^{t_2}$ are partnered if $\prod^{t_1}, \prod^{t_2}$ both meet: a) in the accepted state, b) by mutual authentication with the shared identical session identifier, and c) $\prod^{t_1}, \prod^{t_2}$ are mutual partners in a session.

Adversary: In $\mathbb{C}$, the eCK adversary $\mathcal{A}$ can initiate the query oracles with instance $\prod^t$. The queries that adversary $\mathcal{A}$ can initiate are listed below:

- **Execute** ($\prod^u_{\mathcal{DRN}_u}, \prod^v_{\mathcal{DRN}_v}$, and $\prod^{ca}_{\mathcal{CA}}$). In this query, the adversary $\mathcal{A}$ participates in the authentication process simulated by $\prod^u_{\mathcal{DRN}_u}, \prod^v_{\mathcal{DRN}_v}$, and $\prod^{ca}_{\mathcal{CA}}$, and can yield the complete communication messages among $\prod^u_{\mathcal{DRN}_u}, \prod^v_{\mathcal{DRN}_v}$, and $\prod^{ca}_{\mathcal{CA}}$.
- **Send** ($\prod^t, m$). $\mathcal{A}$ in this query can initiate an active attack. Specifically, $\mathcal{A}$ submits a valid message m to a participating instance $\prod^t$. Accordingly, the $\prod^t$ will give a response to $\mathcal{A}$.
- **SessionKeyReveal** ($\prod^t$). In this query, apart from the session key to be tested, $\mathcal{A}$ can get the other session keys by SessionKeyReveal ($\prod^t$) (and its partner).
- **EphemeralKeyReveal** ($\prod^t$). This query means that the adversary $\mathcal{A}$ can get entities' ephemeral secrets.
- **Corrupt** ($\prod^u_{\mathcal{DRN}_u}$ **and** $\prod^v_{\mathcal{DRN}_v}$). This query means that $\mathcal{A}$ will get the long-term key pair (a_u, A_u) and (a_v, A_v).
- **Corrupt** ($\prod^{ca}_{\mathcal{CA}}$). For this query, the secret value $d_{\mathcal{CA}}$ of $\prod^{ca}_{\mathcal{CA}}$ can be known by $\mathcal{A}$.

Freshness: The confidentiality of a communication is ensured if the adversary $\mathcal{A}$ is unable to discern the session key between the communicating parties $\mathcal{DRN}_u$ and $\mathcal{DRN}_v$ by means of the SessionKeyReveal query.

Test: This is employed to illustrate the semantic security of the session key, $\mathcal{SK}_{uv}$, and the fact that the adversary, $\mathcal{A}$, is only capable of querying the session key once. In light of the aforementioned description of $\mathbb{C}$, it can be inferred that the related instance $\prod^t$ can only be $\prod^u_{\mathcal{DRN}_u}$ and $\prod^v_{\mathcal{DRN}_v}$. In accordance with the formal definition, if the specified instance ($\prod^t$) has not yet established a session key, or if $\prod^t$ is not currently considered fresh, or if the Test ($\prod^t$) query has been conducted previously, the query will immediately conclude and directly output a null value ($\perp$). Conversely, if the aforementioned conditions are not met, the oracle will randomly select a value b, which will be either 0 or 1. If $b = 0$, the Test ($\prod^t$) query will return the actual session key ($\mathcal{SK}_{uv}$). Otherwise, the query will output a random string of the same length as the actual $\mathcal{SK}_{uv}$.

2) Semantic Security: Let $Succ(\mathcal{A})$ denote the advantage of $\mathcal{A}$ correctly guessing b^* of b. The advantage of $\mathcal{A}$ breaking the semantic security of $\mathcal{SK}_{uv}$ in $\mathbb{C}$ can be defined as follows:

$$Adv_{\mathbb{C}}^{\mathcal{A}} = 2\mathcal{P}r[Succ(\mathcal{A})] - 1. \tag{1}$$

Theorem 1. Let $Adv_p^{HECDL}(n)$ and $Adv_p^{HECCDH}(n)$ be defined as the advantage of a PPT adversary $\mathcal{A}$ solving the HECDL and HECCDH hardness problems, respectively. Based on a sequence of queries with oracle, the advantage $Adv_{\mathbb{C}}^{\mathcal{A}}$ of a PPT adversary $\mathcal{A}$ breaking the semantic security of $\mathcal{SK}_{uv}$ in $\mathbb{C}$ is less than:

$$\Delta + \frac{q_h^2 + q_s}{2^{\ell_1 - 1}} + \frac{(q_s + q_e)^2}{p}, \tag{2}$$

where $\Delta = 2(Adv_p^{HECDL}(n) + Adv_p^{HECCDH}(n))$ and q_e (resp., q_s, q_h) denotes number of times $\mathcal{A}$ running Execute-query (resp., Send, Hash-query).

Proof. Here a chain from Game$_1$ to Game$_7$ constitutes a proof process. Let $Succ_i$ mean that $\mathcal{A}$ successfully guesses b in Test-query of Game$_i$, $(i = 1, 2, \ldots, 7)$.

Game$_1$: This game simulates a real attack under the random oracle model. In this game, a bit b is chosen at the beginning. So:

$$Adv_{\mathbb{C}}^{\mathcal{A}} = 2\mathcal{P}r[Succ_1] - 1. \tag{3}$$

Game$_2$: The game model maintains a hash list, designated as Γ_h. Let us consider the following scenario: $\mathcal{A}$ initiates a hash query, designated as $h(\gamma)$, and the hash oracle Θ_h, then takes γ in order to retrieve Γ_h. In the event that a hash value, designated as $h(y)$, can be retrieved from the hash list, Γ_h, the hash oracle Θ_h, will respond with the retrieved hash value. In the event that the aforementioned conditions are not met, the hash oracle Θ_h transmits a randomly generated string ψ to $\mathcal{A}$. Concurrently, the response (γ, ψ) is stored in Γ_h.

By utilizing the known hash list Γ_h in Game$_2$, the adversary $\mathcal{A}$ attempts to discern the value of b by distinguishing it from a random string. In point of fact, the session key $\mathcal{SK}_{uv} = h(\mathcal{C}r_u||N_u||\lambda_{uv}||\mu_{uv}||N_v||\mathcal{C}r_v)$ is derived from the secret values as follows: $\lambda_{uv} = \omega_u\omega_v{\cdot}D, \mu_{uv} = a_u a_v{\cdot}D$. Therefore, in the absence of knowledge regarding the secret values a_u and a_v, it is not possible for $\mathcal{A}$ to compute $\mathcal{SK}_{uv}$ and there is no effective method by which they can distinguish the real session key from a random string, other than through guesswork.

Consequently, the probability of adversary $\mathcal{A}$ winning Game$_2$ does not confer any advantage to $\mathcal{A}$ compared with Game$_1$, that is to say:

$$\mathcal{P}r[Succ_1] = \mathcal{P}r[Succ_2]. \tag{4}$$

Game$_3$: The following section will present a simulation of an active attack in this game. In order to persuade a participant to accept a forged message, the adversary $\mathcal{A}$ initiates a series of Send-query and Hash-query operations. In comparison to the preceding two games, $\mathcal{A}$ may gain an advantage by identifying a collision. In other words, should the following collisions occur, Game$_3$ will be terminated.

1) It is possible to identify collisions in the hash values, and therefore the probability is $\frac{q_h^2}{2^{\ell_1+1}}$, where ℓ_1 denotes the length of the output produced by a hash function.
2) An additional potential collision may be identified by selecting a pair of random numbers with a probability of $\frac{(q_e+q_s)^2}{2p}$.

Consequently, we have:

$$|\mathcal{P}r[Succ_3] - \mathcal{P}r[Succ_2]| \leq \frac{q_h^2}{2^{\ell_1+1}} + \frac{(q_e + q_s)^2}{2p}. \tag{5}$$

Game$_4$: In this game, $\mathcal{A}$ attempts to ascertain $\mathcal{C}r_i, A_i, \mathcal{SK}_{uv}$, and $\mathcal{VK}_{uv}$ without utilizing a Hash-query. Thus, the following can be inferred:

$$|\mathcal{P}r[Succ_4] - \mathcal{P}r[Succ_3]| \leq \frac{q_s}{2^{\ell_1}}. \tag{6}$$

Game$_5$: In this game, the adversary $\mathcal{A}$ interacts with the EphemeralKeyReveal ($\prod^t$) oracle and the SessionKeyReveal ($\prod^t$) oracle [35], obtaining outdated session keys $(\mathcal{SK}_{uv})_{outdated}$ and the temporary secrets ω_u and ω_v. Thus, $\mathcal{A}$ attempts to corrupt the authentication. However, $\mathcal{A}$ is unable to reveal ω_u^* and N_i from $\mathcal{DRN}_i$'s memory. Alternatively, $\mathcal{A}$ may attempt to identify a collision in the hash values.

This results in the following inequality:

$$|\mathcal{P}r[Succ_5] - \mathcal{P}r[Succ_4]| \leq \frac{q_h^2}{2^{\ell_1+1}}. \tag{7}$$

Game$_6$: In this game, the adversary $\mathcal{A}$ interacts with the Corrupt ($\prod_{\mathcal{CA}}^{ca}$) oracle, thereby acquiring the private key $d_{\mathcal{CA}}$ of Certifier's Authority $\mathcal{CA}$. Although $\mathcal{A}$ has gained access to d_{CA}, the fact that no PPT solution can be used to break the HECDL problem means that $\mathcal{A}$ cannot extract a_i from $\mathcal{C}r_i$. Therefore, we can conclude that:

$$|\mathcal{P}r[Succ_6] - \mathcal{P}r[Succ_5]| \leq Adv_p^{HECDL}(n). \tag{8}$$

Game$_7$: In the final stage of the game, $\mathcal{A}$ is unable to initiate any queries and must instead attempt to compute the session key $\mathcal{SK}_{uv}$. Similarly, given that no PPT solution can be employed to solve the HECCDH hardness problem, $\mathcal{A}$ has a negligible advantage $Adv_p^{HECCDH}(n)$ to obtain $\lambda_{uv} = \omega_u\omega_v{\cdot}D$ and $\mu_{uv} = a_u a_v{\cdot}D$. Therefore:

$$|\mathcal{P}r[Succ_7] - \mathcal{P}r[Succ_6]| \leq Adv_p^{HECCDH}(n). \tag{9}$$

Now, it is evident that $\mathcal{A}$ does not possess any discernible advantage over $\frac{1}{2}$, and so $\mathcal{P}r[Succ_7] = \frac{1}{2}$. Consequently, from (3) to (9), by employing the triangular inequality, we obtain the following deduction:

$$\begin{aligned} Adv_{\mathbb{C}}^{\mathcal{A}} &= 2\mathcal{P}r[Succ_1] - 1 \\ &= 2\mathcal{P}r[Succ_7] - 1 - 2(\mathcal{P}r[Succ_7] - \mathcal{P}r[Succ_1]) \\ &= 2\mathcal{P}r[Succ_7] - 1 - 2\sum_{i=1}^{6}(\mathcal{P}r[Succ_{i+1}] - \mathcal{P}r[Succ_i]) \\ &\leq \Delta + \frac{q_h^2 + q_s}{2^{\ell_1 - 1}} + \frac{(q_e + q_s)^2}{p}. \end{aligned}$$

Therefore, it can be concluded that the adversary $\mathcal{A}$ has negligible advantage $Adv_{\mathbb{C}}^{\mathcal{A}}$ in breaking the semantic security of $\mathcal{SK}_{uv}$.

□

C. Informal Security Analysis

*1) **Mutual Authentication**:* The proposed scheme can provide mutual authentication. Suppose if $\Psi_1 = \{\mathcal{E}_{A_v}(ID_u, N_u, A_u, \mathcal{C}r_u), \mathcal{X}_u, \Omega_u, \Omega_u^\circ, \Lambda_u\}$ to $\mathcal{DRN}_v$.

After receiving Ψ_1, $\mathcal{DRN}_v$ will performs the following computations:

- First, it decrypts the message Ψ_1 using the private key a_v to get $ID_u, N_u, A_u, \mathcal{C}r_u, \mathcal{X}_u, \Omega_u, \Omega_u^\circ, \Lambda_u$. Meanwhile, it uses the formula to calculate the temporary public key Ω_u^*: $\Omega_u^* = \Omega_u^\circ - \Omega_u$.
- Check the certificate of the source drone by applying the following conditions:

$$w_{\mathcal{CA}} + h(ID_u||\mathcal{X}_u)\cdot\mathcal{X}_u \stackrel{?}{=} \mathcal{C}r_u\cdot D.$$

- Attempt to verify the signature, and to check the relationship, verify the following conditions:

$$\mathcal{C}r_u\cdot D + h(ID_u||\mathcal{C}r_u||A_u||\mathcal{X}_u||N_u)\cdot(\Omega_u^\circ + A_u) \stackrel{?}{=} \Lambda_u\cdot D.$$

Consequently, the veracity of the identity of $\mathcal{DRN}_u$ is confirmed by $\mathcal{DRN}_v$.

In a similar manner, upon the transmission of the message, $\mathcal{DRN}_u$ is able to ascertain the veracity of the identity of $\mathcal{DRN}_v$ by utilizing the following method:

- First, $\mathcal{DRN}_u$ decrypts the message Ψ_2 with the private key a_u to get $NON_{ev}, \Lambda_v, \mathcal{C}r_v, \mathcal{VK}_{uv}, ID_v, \Omega_v^\circ, A_v, \mathcal{X}_v$. Then, it proceeds to decrypt $(N_v, ID_v) = \mathcal{D}_{\omega_u^*}(NON_{ev})$ to get and check the freshness of N_v.
- Check the certificate of the drone $\mathcal{DRN}_v$ by computing the following conditions:

$$w_{\mathcal{CA}} + h(ID_v||\mathcal{X}_v)\cdot\mathcal{X}_v \stackrel{?}{=} \mathcal{C}r_v\cdot D.$$

- Eventually, $\mathcal{DRN}_u$ verifies the signature, and to check the relationship, verifying the following conditions:

$$\mathcal{C}r_v\cdot D + h(ID_v||\mathcal{C}r_v||A_v||\mathcal{X}_v||N_v)\cdot(\Omega_v^\circ + A_v) \stackrel{?}{=} \Lambda_v\cdot D.$$

In summary, our scheme allows for mutual authentication.

*2) **Forward Secrecy**:* If the adversary $\mathcal{A}$ successfully attacks the certifier's authority $\mathcal{CA}$, and obtains $\mathcal{CA}$'s private key $d_{\mathcal{CA}}$, our scheme is still forward secure.

The adversary wants to compute:

$$\mathcal{SK}_{uv} = h(\mathcal{C}r_u||N_u||\lambda_{uv}||\mu_{uv}||N_v||\mathcal{C}r_v).$$

This requires $\mathcal{A}$ to obtain the values of parameters $\mathcal{C}r_u, N_u, \lambda_{uv}, \mu_{uv}, N_v, \mathcal{C}r_v$. However, even if the adversary has been in control of $\mathcal{CA}$'s private key $d_{\mathcal{CA}}$, $\mathcal{A}$ is still unable to restore the value of $\mathcal{C}r_u$ or $\mathcal{C}r_v$ because the value of the random salt f_u or f_v can only be extracted from the equation:

$$\mathcal{X}_u = (f_u + a_u)\cdot D,$$
$$\mathcal{X}_v = (f_v + a_v)\cdot D.$$

This is equivalent to solving the HECDLP at least twice. Similarly, without the values of the private keys a_u and a_v, the other parameters are also unobtainable. Consequently, the proposed scheme is forward secure, in that all preceding session keys remain secure even in the event of a long-term private key compromise at the certifier's authority $\mathcal{CA}$.

*3) **MITM Attack**:* Suppose the adversary $\mathcal{A}$ intends to alter the message Ψ_1 transmitted between the two drones $\mathcal{DRN}_u$ and $\mathcal{DRN}_v$, the message $\Psi_1 = \{\mathcal{E}_{A_v}(ID_u, N_u, A_u, \mathcal{C}r_u), \mathcal{X}_u, \Omega_u, \Omega_u^\circ, \Lambda_u\}$ comprises of multiple parameters, some of which are encrypted. Thus, to be successful in its malicious attack, the adversary $\mathcal{A}$ will need to access the values of Λ_u:

$$\Lambda_u = \mathcal{C}r_u + h(ID_u||\mathcal{C}r_u||A_u||\mathcal{X}_u||N_u)(\omega_u + \omega_u^* + a_u).$$

Here, N_u only can be obtained by decrypting message Ψ_1 with a_v. Similarly, $\mathcal{C}r_u$ and other parameters are encrypted. In other words, $\mathcal{A}$ must have the values of a_u and a_v, which could only be obtained from $A_u = a_u\cdot D$ and $A_v = a_v\cdot D$. Besides, $\mathcal{A}$ needs to calculate $(\omega_u + \omega_u^*)$ by $\Omega_u^\circ = (\omega_u + \omega_u^*)\cdot D$.

Performing such mathematical maneuvers is impracticable since it is equivalent to solving the HECDLP three times. As it is difficult, in consequence, $\mathcal{A}$ is impossible to solve the problem, and so the proposed scheme can resist the Man-in-the-Middle Attacks.

*4) **Drone Impersonation Attack**:* Assuming adversary $\mathcal{A}$ attempts to impersonate $\mathcal{DRN}_v$ to generate the message:

$$\Psi_2 = \{\mathcal{E}_{A_u}(NON_{ev}, \Lambda_v, \mathcal{C}r_v, \mathcal{VK}_{uv}), ID_v, \Omega_v^\circ, A_v, \mathcal{X}_v\}.$$

Such attempts, in turn, would require a great deal of computation to produce the variable Λ_v and $\mathcal{VK}_{uv}$. The mathematical formulae involved are as follows:

1) $\mathcal{VK}_{uv} = h(N_u||N_v||\mathcal{SK}_{uv})$

Here, to approach the value of $\mathcal{VK}_{uv}$, $\mathcal{A}$ is required to compute the value of N_u and $\mathcal{SK}_{uv}$. For the reason that N_u is encrypted by A_v in the message Ψ_1 in the open channel, only when $\mathcal{A}$ have the private key of $\mathcal{DRN}_v$ can he get the proper value of N_u to compute $\mathcal{VK}_{uv}$, which means $\mathcal{A}$ is need to solve the HECDLP according to $A_v = a_v\cdot D$. Thus, $\mathcal{A}$ is impossible to fabricate $\mathcal{VK}_{uv}$.

2) $\Lambda_v = \mathcal{C}r_v + h(ID_v||\mathcal{C}r_v||A_v||\mathcal{X}_v||N_v)(\omega_v + \omega_v^* + a_v)$

To execute such calculation, it is vital for $\mathcal{A}$ to extract the value of a_u, a_v, and $(\omega_v + \omega_v^*)$ to get the necessary elements from the following equations:

$$A_u = a_u\cdot D,$$

$$A_v = a_v\cdot D,$$

$$\Omega_v^\circ = (\omega_v + \omega_v^*)\cdot D.$$

To solve such an equation is equivalent to solving the HECDLP three times, which proves it is unable for $\mathcal{A}$ to falsify $\mathcal{DRN}_v$'s signature Λ_v.

On balance, if the adversary $\mathcal{A}$ attempts to impersonate $\mathcal{DRN}_v$, complex mathematical operations are required. This mathematical operation is equivalent to calculating the discrete logarithm problem of hyperelliptic curves at least four times. Therefore, the proposed scheme provides protection against drone impersonation attacks.

*5) **Replay Attack**:* We assume that the adversary $\mathcal{A}$ aims to intercept the communication between $\mathcal{DRN}_u$ and $\mathcal{DRN}_v$, attempting to capture and replay the message Ψ_1 in the open channel:

$$\Psi_1 = \{\mathcal{E}_{A_v}(ID_u, N_u, A_u, \mathcal{C}r_u), \mathcal{X}_u, \Omega_u, \Omega_u^{\circ}, \Lambda_u\}.$$

After receiving the replayed message from $\mathcal{A}$, $\mathcal{DRN}_v$ decrypts the message with its own private key to get the nonce N_u and checks its freshness and legitimacy. Since this message is a replayed message, its nonce N_u cannot pass $\mathcal{DRN}_v$'s test, so the attack is unsuccessful.

*6) **Node Capture Attack**:* Suppose adversary $\mathcal{A}$ wants to generate the following session key:

$$\mathcal{SK}_{uv} = h(\mathcal{C}r_u||N_u||\lambda_{uv}||\mu_{uv}||N_v||\mathcal{C}r_v).$$

It is now necessary to consider the capabilities possessed by the adversary. It is possible for $\mathcal{A}$ to capture and obtain a UAV's memory including the long-term private key, as well as other parameters, belonging to one side the communication parties $\mathcal{DRN}_u$ and $\mathcal{DRN}_v$. The following scenario illustrates the subsequent actions taken by $\mathcal{A}$ upon obtaining different key values:

1) $\mathcal{DRN}_u$'s private key a_u is leaked

If $\mathcal{DRN}_u$'s private key a_u is leaked, the aforementioned key will permit access to the encrypted message that was transmitted via the public channel. In this case, $\mathcal{A}$ can acquire $NON_{ev}, \Lambda_v, \mathcal{C}r_v, \mathcal{VK}_{uv}$ by computing:

$$(NON_{ev}, \Lambda_v, \mathcal{C}r_v, \mathcal{VK}_{uv}), ID_v, \Omega_v^{\circ}, A_v, \mathcal{X}_v = \mathcal{D}_{a_u}(\Psi_2).$$

To generate the session key $\mathcal{SK}_{uv}$, $\mathcal{A}$ must also solve the values of $\mathcal{C}r_u, N_u, \lambda_{uv}, \mu_{uv}$, and N_v. The certificate $\mathcal{C}r_u$ can be calculated by equation

$$\mathcal{C}r_u = d_{\mathcal{CA}} + (f_u + a_u)h(ID_u||\mathcal{X}_u),$$

where $d_{\mathcal{CA}}$ is the $\mathcal{CA}$'s private key, and the only way to get it is by using equation $w_{\mathcal{CA}} = d_{\mathcal{CA}} \cdot D$. To perform such a computation is equivalent to solving the HECDLP, which is far too complex to be solved by the adversary $\mathcal{A}$.

Then, in the broadcast channel, it is not possible for $\mathcal{A}$ to ascertain the value of N_u, as the message Ψ_1 ought to be decrypted by a_v. Also, to get the value of N_v, $\mathcal{A}$ needs to perform the calculation $(N_v, ID_v) = \mathcal{D}_{\omega_u^*}(NON_{ev})$, the value ω_u^* needs to be extracted from the relation $\omega_u^* \cdot D = \Omega_u^* = \Omega_u^{\circ} - \Omega_u$, and performing such a computation is tantamount to solving the HECDLP. With respect to λ_{uv} and μ_{uv}, although $\mathcal{A}$ can obtain the value of μ_{uv} through equation $\mu_{uv} = a_u \cdot A_v = a_u a_v \cdot D$, it is unable to compute the value of λ_{uv} because it lacks the requisite value of ω_v.

2) $\mathcal{DRN}_v$'s private key a_v is leaked

If $\mathcal{DRN}_v$'s private key a_v is leaked, $ID_u, N_u, \Lambda_u, \mathcal{C}r_u$ can be acquired by decrypting:

$$(ID_u, N_u, A_u, \mathcal{C}r_u), \mathcal{X}_u, \Omega_u, \Omega_u^{\circ}, \Lambda_u = \mathcal{D}_{a_v}(\Psi_1).$$

In order to generate the session key $\mathcal{SK}_{uv}$, $\mathcal{A}$ must solve the values of $\mathcal{C}r_v, \lambda_{uv}, \mu_{uv}$, and N_v. Except for μ_{uv}, the remaining parameters are not available, so $\mathcal{A}$ cannot compute $\mathcal{SK}_{uv}$ under this circumstance. Hence, in our proposed scheme, even if it is assumed that the adversary has knowledge of one of the long-term private keys of the two UAVs, this adversary cannot retrieve the session key $\mathcal{SK}_{uv}$ unless they can effectively break the hardness of the HECDL problem.

*7) **ESL Attack**:* If either $\mathcal{DRN}_u$'s or $\mathcal{DRN}_v$'s temporary private key ω_u or ω_v is leaked, the adversary $\mathcal{A}$ will not be able to unravel any of the parameters that are necessary to generate $\mathcal{SK}_{uv}$ except for λ_{uv}. Therefore, it can be concluded that the proposed scheme is also resilient against ESL attacks.

*8) **Malicious Drone Deployment Attack**:* Consider a scenario where the adversary $\mathcal{A}$ attempts to infiltrate an established network by deploying a fraudulent drone, denoted as $\mathcal{DRN}_{fraud}$. The adversary's modus operandi involves:

1) Assigning a fake identity, ID_{fraud}, and a randomly generated private key, a_{fraud}.
2) Deriving the public identity as ID_{fraud}, and computing: $A_{fraud} = a_{fraud} \cdot D$.
3) Choosing f_{fraud} and computing $\mathcal{X}_{fraud}$: $\mathcal{X}_{fraud} = (f_{fraud} + a_{fraud}) \cdot D$.
4) Generating a fake certificate for ID_{fraud}: $\mathcal{C}r_{fraud} = d_{fraud} + (f_{fraud} + a_{fraud})h(ID_{fraud}||\mathcal{X}_{fraud})$.
5) Subsequently, loading the parameters set $(ID_{fraud}, \mathcal{C}r_{fraud}, A_{fraud}, a_{fraud}, \mathcal{X}_{fraud})$ into $\mathcal{DRN}_{fraud}$'s memory.

In practicality, to authenticate $\mathcal{DRN}_{fraud}$ with a legitimate certificate, the intruder needs $d_{\mathcal{CA}}$, derivable from $w_{\mathcal{CA}} = d_{\mathcal{CA}} \cdot D$. However, manipulating this relationship is as challenging as solving an HECDLP. Consequently, among other safeguards, our scheme defends the system against Malicious Drone Deployment Attacks.

*9) **Anonymity Preservation**:* In our proposed scheme, the principle of anonymity is upheld. The ciphertext Ψ_1 does not directly contain the information. The drone identity can only be derived from the ciphertext as follows:

$$(ID_u, N_u, A_u, \mathcal{C}r_u), \mathcal{X}_u, \Omega_u, \Omega_u^{\circ}, \Lambda_u = \mathcal{D}_{a_v}(\Psi_1).$$

In order to obtain the drone identity, the private key, a_v, must be used. This requires the solution of the hyperelliptic curve discrete logarithmic problem once. Furthermore, it is challenging to retrieve ID_u from Λ_u or $\mathcal{C}r_u$ due to the unidirectional nature of the hash function. Consequently, it can be concluded from the aforementioned argument that the proposed scheme effectively guarantees anonymity.

*10) **Cloning Attack**:* In a drone cloning attack, an adversary attempts to replicate or mimic the identity of a legitimate drone, $\mathcal{A}$ obtaining the identity information of the legitimate drone, which includes the drone's identifier ID_i, certificate $\mathcal{C}r_i$, and other authentication credentials.

Nevertheless, as a consequence of the fact that in our scheme a provisional set of session keys ω_i and ω_i^* is selected for each session, such an update renders our scheme resistant to cloning attacks.

*11) **De-synchronization Attack**:* During the communication process between the sender $\mathcal{DRN}_u$ and the receiver $\mathcal{DRN}_v$, it is possible for $\mathcal{A}$ to prevent the simultaneous updating of secret information. In order to prevent this attack, the receiving drone $\mathcal{DRN}_v$ is equipped with the capability to

TABLE III
COMPARISON OF SECURITY ATTRIBUTES

Security Attribute	Our Scheme	Muhammad *et al.* [17]	Gope *et al.* [36]	Yahuza *et al.* [37]	Bera *et al.* [38]	Huang *et al.* [39]
Mutual Authentication	✓	✓	✓	✓	✓	✓
Forward Secrecy	✓	✗	✓	✗	✗	✓
Anonymity Preservation	✓	✓	✓	✗	✓	✓
MITM Attack	✓	✓	✓	✓	✓	✓
Replay Attack	✓	✓	✓	✓	✓	✓
Drone Impersonation Attack	✓	✓	✓	✓	✓	✓
Malicious Drone Deployment Attack	✓	✓	✓	✓	✓	✓
Cloning Attack	✓	✗	✓	✗	✗	✓
De-synchronization Attack	✓	✓	✓	$\mathcal{O}$	$\mathcal{O}$	✓
DoS Attack	✓	✓	$\mathcal{O}$	$\mathcal{O}$	$\mathcal{O}$	$\mathcal{O}$
Node Capture Attack	✓	✗	✗	✓	✗	✓
ESL Attack	✓	✗	✓	✓	✓	✓

✓: Provided & Resistant; ✗:Vulnerable; $\mathcal{O}$: Not Provided.

store the previous message in the database, accompanied by its nonce. In the event that $\mathcal{A}$ transmits the message Ψ_1 to $\mathcal{DRN}_v$, the latter proceeds to decrypt the parameters, and assesses the freshness of N_u. This process enables the receiving drone to evade the de-synchronization attack.

12) ***Denial-of-Service (DoS) Attack****:* It should be noted that the solution is not susceptible to DoS attacks. To illustrate, if $\mathcal{DRN}_v$ transmits Ψ_2, where $\Psi_2 = \{\mathcal{E}_{A_u}(NON_{ev}, \Lambda_v, \mathcal{C}r_v, \mathcal{VK}_{uv}), ID_v, \Omega_v^\circ, A_v, \mathcal{X}_v\}$ to $\mathcal{DRN}_u$, the initial step is to decrypt Ψ_2 and then NON_{ev} to get N_v, subsequently verifying the freshness of the nonce N_v. If the nonce is no longer fresh, the DoS attack is effectively resisted.

TABLE IV
THE RUNNING TIME FOR EACH OPERATION

Item	Descriptions	Time (ms)
T_{hm}	Hyperelliptic curve scalar multiplication	4.8×10^{-1}
T_{em}	Elliptic curve scalar multiplication	1.206
T_{ea}	Elliptic curve addition	2.88×10^{-2}
$T_{\mathcal{AE}}$	The time of AES's encryption	3.53×10^{-2}
T_h	Execute a secure hash function	1.2×10^{-2}
$T_{\mathcal{E}}$	The time of HECC's encryption	1.1×10^{-3}
$T_{\mathcal{D}}$	The time of HECC's decryption	3.5×10^{-4}
$T_{\mathcal{P}}$	Secure PUF operation	1.12
T_{fe}	Fuzzy extractor generation/reconstruction	3.46

VII. PERFORMANCE ANALYSES

In this section, we conduct a comprehensive evaluation of our proposed scheme against six state-of-the-art solutions.

A. Comparative Security Analysis

The security resilience of an AKA protocol is paramount for mission-critical UAV operations. A comparative analysis of our proposed scheme against the benchmark solutions, based on twelve critical security attributes, is summarized in Table III.

As demonstrated, our proposed scheme satisfies all the evaluated security requirements, offering a holistic security guarantee. In contrast, the baseline schemes exhibit critical vulnerabilities that compromise their suitability for secure deployment.

A recurring and fundamental weakness across six schemes, including those by Muhammad *et al.* [17], Yahuza *et al.* [37], and Bera *et al.* [38], is the lack of forward secrecy. This implies that the compromise of a single long-term key allows an adversary to retrospectively decrypt all past communications, catastrophically compromising mission history.

More alarmingly, the majority of the compared schemes are vulnerable to physical attacks, a primary threat for deployed UAVs. Protocols by Muhammad *et al.*, Yahuza *et al.*, and Bera *et al.* are susceptible to both node capture or cloning attacks because their security relies solely on digitally stored secrets that can be extracted and replicated. Even the PUF-based scheme by Gope *et al.* [36], while resistant to cloning, is critically flawed; by storing a long-term secret key in memory, it creates a single point of failure that bypasses the PUF's physical security once the device is captured. Other significant issues include a complete lack of anonymity in the work of Yahuza *et al.*, which transmits real identities in plaintext, and a specific vulnerability to ESL attacks in the scheme by Muhammad *et al*.

Therefore, each baseline scheme suffers from at least one critical vulnerability, ranging from inadequate privacy to complete compromise upon physical capture. Our proposed scheme comprehensively addresses these shortcomings, presenting a robust and reliable solution for UAV networks.

B. Theoretical Overhead Analysis

We evaluate the schemes' practicality for resource-constrained UAVs by analyzing theoretical computational cost and communication cost.

1) Computational Cost: To establish a fair and reproducible comparison, we benchmarked the execution times of the dominant cryptographic operations. To uniformly simulate the constrained computational resources of a typical UAV, these operations are conducted on a machine equipped with an `Intel Core i7-4510U` CPU (2.0 GHz), 8 GB of RAM, and running a `Windows 7` operating system. The resulting average execution times, which form the basis for

TABLE V
PERFORMANCE COMPARISON OF DIFFERENT SCHEMES WITH VARYING NUMBERS OF UAVS

Scheme	No. of UAVs	Application Layer Performance		Network Layer Performance		
		Auth. Succ. Rate (%)	Avg. Auth. Delay (ms)	Packet Loss Rate (%)	Avg. Comm. Delay (ms)	Avg. Throughput (kbps)
Our Scheme	5	100.000	6.67148	0.00000	0.68357	0.87204
	15	99.9237	7.27594	0.05724	0.83872	2.28900
	25	99.9138	7.77726	0.08627	0.93891	4.05189
Muhammad *et al.* [17]	5	100.000	6.59120	0.00000	0.71065	0.88091
	15	99.7728	7.30808	0.07581	1.00349	2.33418
	25	99.8016	7.63384	0.07355	1.10129	4.00966
Gope *et al.* [36]	5	100.000	30.0938	0.00000	0.95821	1.04966
	15	99.8569	30.7835	0.09549	1.26343	2.94581
	25	99.8491	31.0115	0.07191	1.29891	4.89088
Yahuza *et al.* [37]	5	100.000	17.7889	0.00000	0.79176	1.19931
	15	99.6319	18.5191	0.12297	1.07505	2.93722
	25	99.8474	18.8452	0.05817	1.18630	5.52299
Bera *et al.* [38]	5	100.000	17.0820	0.00000	0.89453	1.59617
	15	99.4328	17.7644	0.18956	1.14970	3.67540
	25	99.7273	18.6204	0.09101	1.44478	7.07274
Huang *et al.* (1) [39]	5	100.000	12.3630	0.00000	1.06308	1.46572
	15	99.6403	13.0479	0.16017	1.34930	3.64326
	25	99.8466	13.3611	0.05844	1.43608	6.66380
Huang *et al.* (2) [39]	5	100.000	9.52714	0.00000	0.77340	1.12878
	15	99.7393	10.2083	0.14912	1.05441	3.05490
	25	99.8946	10.5075	0.09839	1.15280	5.40626

TABLE VI
TIME COMPLEXITY OF COMPUTATIONAL COST

Schemes	Formal Computational Cost
Muhammad *et al.* [17]	$14T_{hm} + 11T_h + 3T_{\mathcal{E}} + 3T_{\mathcal{D}}$
Gope *et al.* [36]	$10T_h + 2T_{\mathcal{P}} + 2T_{fe}$
Yahuza *et al.* [37]	$6T_{em} + 6T_h + 10T_{ea}$
Bera *et al.* [38]	$12T_{em} + 12T_h + 4T_{ea}$
Huang *et al.* (1) [39]	$8T_{em} + 9T_h + 2T_{ea} + 2T_{\mathcal{P}}$
Huang *et al.* (2) [39]	$16T_{em} + 14T_h + 4T_{ea} + 2T_{\mathcal{P}} + 2T_{\mathcal{AE}}$
Our scheme	$15T_{hm} + 10T_h + 3T_{\mathcal{E}} + 3T_{\mathcal{D}}$

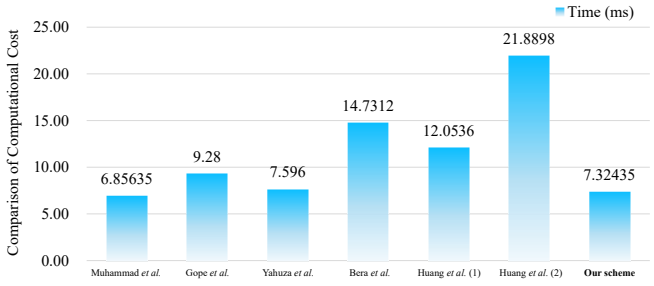

Fig. 4. Comparison of baselines' computational cost.

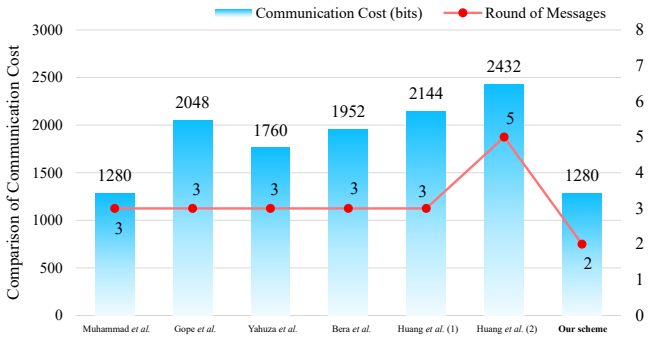

Fig. 5. Comparison of baselines' communication cost.

our entire theoretical analysis, are summarized in Table IV. These timings are consistent with values reported in related works [17], [40]–[42]. Key operations include hyper-elliptic curve scalar multiplication (HCSM) at 0.48 ms, elliptic curve scalar multiplication (ECSM) at 1.206 ms, and secure PUF operations at 1.12 ms.

By applying these benchmarked timings to the operational counts for each protocol, as detailed in Table VI, we derive the total computational cost for a complete authentication session. The aggregated results, visualized in Fig. 4, demonstrate our proposed scheme's superiority. While the scheme by Muhammad *et al.* [17] is marginally faster by approximately 0.5 ms, it is critically vulnerable to both ESL and Type-I node capture attacks. In contrast, our scheme achieves the second-lowest computational cost while providing robust security, confirming its excellent balance of efficiency and safety for latency-sensitive applications.

2) Communication Cost: To compare communication cost, we quantify the overhead based on the number of messages exchanged and their sizes. For each scheme, we calculate the total length and round of messages involved in the authentication and key agreement process. From Fig. 5, we see that our scheme requires the fewest rounds of messages and communication cost in bits, saving nearly 50% of communication cost compared with Huang *et al.*'s scheme (2) [39], thereby reducing computational cost, memory requirements, and energy consumption. Notably, our scheme only takes two rounds of communication to achieve AKA after optimization. It demonstrates our scheme's efficiency in minimizing communication, which underscores the suitability for deployment in bandwidth-constrained and power-sensitive SAR missions.

C. Experimental Validation and Comparative Analysis

To assess the practical performance and scalability of our proposed scheme, a series of simulations are conducted using the `NS-3.26` network simulator.[1] The simulations are executed within a virtualized `CentOS 7` environment hosted on a machine equipped with an `Intel Core i7-4790` CPU operating at 3.60 GHz and 16.0 GB of RAM. Our simulation models a dynamic ad-hoc network within a 2000 m × 2000 m area, where the number of UAVs was systematically varied from 5 to 25 to evaluate each protocol's performance under increasing network density. The communication stack is configured based on the *IEEE 802.11b* standard with the OLSR protocol managing multi-hop routing. The key performance indicators for our scheme and the benchmark solutions are meticulously recorded and are presented in Table V.

The experimental results unequivocally demonstrate the superior comprehensive performance of our proposed scheme. As shown in Table V, our protocol consistently delivers an exceptionally low average authentication delay, remaining stable between 6.7 ms and 7.8 ms across all tested network scales. This sustained sub-10 ms performance not only satisfies but significantly exceeds the stringent requirements for time-critical applications defined in standards including *ITU-T Y.2066* and *ITU-T F.749.10*. Furthermore, its reliability is underscored by an authentication success rate that consistently surpasses 99.9%, the highest among all schemes in high-density scenarios, and a negligible packet loss rate, confirming its robustness and compliance with the high-availability expectations of *IEEE Std 1936.1-2021* and *IEEE Std 1609.2-2022*.

Most baseline schemes exhibit significant latency overhead. The schemes by Gope *et al.* [36], Yahuza *et al.* [37], Bera *et al.* [38], and Huang *et al.* [39] exhibit authentication delays that are 1.5 to 4 times higher than our scheme, rendering them less suitable for real-time control loops in dynamic UAV environments. Moreover, the higher network throughput observed in these baseline schemes is not an indicator of superior efficiency but rather of greater communication overhead, as the authentication process itself should consume minimal bandwidth to maximize resources for the actual mission data. Our scheme's lower throughput is a testament to its lightweight system design.

VIII. Conclusion

AKA mechanisms are crucial for ensuring the security of UAV communications. This paper has presented a new HECC-based AKA scheme for resource-constrained UAV systems, which ensures robust UAV mutual authentication with precise identity verification while facilitating secure session key establishment to maintain communication confidentiality. By leveraging HECC, the proposed scheme has provided robust security against threats including ESL attacks and node capture attacks under the eCK adversarial model. The scheme's security analysis has demonstrated its resilience against eCK adversarial threats. Performance comparisons with six schemes have further demonstrated its efficiency, with significant reductions in computational cost and communication cost. It has been shown that fewer communication rounds can still achieve a balance between security assurance and operational efficiency in resource-limited aerial environments.

[1]Our code is available at `https://github.com/NoWall-572/NS-3_Communication_Simulation_for_AKAs`.

References

[1] Y. Zeng, Q. Wu, and R. Zhang, "Accessing from the sky: A tutorial on UAV communications for 5G and beyond," *Proc. IEEE*, vol. 107, no. 12, pp. 2327–2375, Dec. 2019.

[2] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, May 2016.

[3] G. Gui, M. Liu, F. Tang, N. Kato, and F. Adachi, "6G: Opening New horizons for integration of comfort, security, and intelligence," *IEEE Wireless Commun.*, vol. 27, no. 5, pp. 126–132, Oct. 2020.

[4] N. Gao, L. Liang, D. Cai, X. Li, and S. Jin, "Coverage control for UAV swarm communication networks: A distributed learning approach," *IEEE Internet Things J.*, vol. 9, no. 20, pp. 19 854–19 867, Oct. 2022.

[5] S. Qi, B. Lin, Y. Deng, X. Chen, and Y. Fang, "Minimizing maximum latency of task offloading for multi-UAV-assisted maritime search and rescue," *IEEE Trans. Veh. Technol.*, vol. 73, no. 9, pp. 13 625–13 638, Sep. 2024.

[6] J. Mu, R. Zhang, Y. Cui, N. Gao, and X. Jing, "UAV Meets integrated sensing and communication: Challenges and future directions," *IEEE Commun. Mag.*, vol. 61, no. 5, pp. 62–67, May 2023.

[7] N. Gao, Z. Qin, X. Jing, Q. Ni, and S. Jin, "Anti-Intelligent UAV jamming strategy via deep q-networks," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 569–581, Jan. 2020.

[8] P. Tedeschi *et al.*, "Privacy-Aware remote identification for unmanned aerial vehicles: Current solutions, potential threats, and future directions," *IEEE Trans. Ind. Inf.*, vol. 20, no. 2, pp. 1069–1080, Feb. 2024.

[9] W. Zhang, X. Liang, Q. Deng, F. Shu, Z. Zhang, L. Nie, and S. Yan, "Joint trajectory and beamforming optimization for IRS-assisted multi-antenna UAV covert communications with a finite blocklength," *IEEE Trans. Green Commun. Netw.*, vol. 10, pp. 426–439, Oct. 2026.

[10] Z. Yu *et al.*, "Cybersecurity of unmanned aerial vehicles: A survey," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 39, no. 9, pp. 182–215, Sep. 2024.

[11] R. Aissaoui, J.-C. Deneuville, C. Guerber, and A. Pirovano, "A survey on cryptographic methods to secure communications for UAV traffic management," *Veh. Commun.*, vol. 44, p. 100661, Dec. 2023.

[12] S. Yu, J. Lee, A. K. Sutrala, A. K. Das, and Y. Park, "LAKA-UAV: Lightweight authentication and key agreement scheme for cloud-assisted unmanned aerial vehicle using blockchain in flying ad-hoc networks," *Comput. Netw.*, vol. 224, p. 109612, Apr. 2023.

[13] H. Tan, W. Zheng, and P. Vijayakumar, "Secure and efficient authenticated key management scheme for UAV-assisted infrastructure-less iovs," *IEEE Trans. Intell. Transport. Syst.*, vol. 24, no. 6, pp. 6389–6400, Jun. 2023.

[14] M. Gates, "Flight path compromised: New research reveals drone vulnerability," Security Management Magazine, Jun. 2023, [Online].

[15] J. Cui *et al.*, "A practical and provably secure authentication and key agreement scheme for UAV-assisted vanets for emergency rescue," *IEEE Trans. Netw. Sci. Eng.*, vol. 11, no. 2, pp. 1454–1468, Mar. 2024.

[16] B. Pratap, A. Singh, and P. S. Mehra, "REHAS: Robust and efficient hyperelliptic curve-based authentication scheme for internet of drones," *Concurr. Comput.*, vol. 37, no. 3, p. e8333, Feb. 2025.

[17] M. A. Khan *et al.*, "An efficient and secure certificate-based access control and key agreement scheme for flying ad-hoc networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 5, pp. 4839–4851, May 2021.

[18] P. Gope and B. Sikdar, "An efficient privacy-preserving authenticated key agreement scheme for edge-assisted internet of drones," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13 621–13 630, Nov. 2020.

[19] H. Shakhatreh *et al.*, "Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges," *IEEE Access*, vol. 7, pp. 48 572–48 634, 2019.

[20] G. K. Pandey *et al.*, "Security threats and mitigation techniques in UAV communications: A comprehensive survey," *IEEE Access*, vol. 10, pp. 112 858–112 897, 2022.

[21] C. Pu, A. Wall, K.-K. R. Choo, I. Ahmed, and S. Lim, "A lightweight and privacy-preserving mutual authentication and key agreement protocol for internet of drones environment," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9918–9933, Jun. 2022.

[22] Y. Zhang, D. He, L. Li, and B. Chen, "A lightweight authentication and key agreement scheme for internet of drones," *Comput. Commun.*, vol. 154, pp. 455–464, Mar. 2020.
[23] S. U. Jan, I. A. Abbasi, and F. Algarni, "A key agreement scheme for IoD deployment civilian drone," *IEEE Access*, vol. 9, pp. 149 311–149 321, 2021.
[24] J. Zhang, J. Cui, H. Zhong, I. Bolodurina, and L. Liu, "Intelligent drone-assisted anonymous authentication and key agreement for 5G/B5G vehicular ad-hoc networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 4, pp. 2982–2994, Oct. 2021.
[25] D. Wang, Y. Cao, K.-Y. Lam, Y. Hu, and O. Kaiwartya, "Authentication and key agreement based on three factors and PUF for UAV-assisted post-disaster emergency communication," *IEEE Internet Things J.*, vol. 11, no. 11, pp. 20 457–20 472, Jun. 2024.
[26] Y. Kirsal Ever, "A secure authentication scheme framework for mobile-sinks used in the internet of drones applications," *Comput. Commun.*, vol. 155, pp. 143–149, Apr. 2020.
[27] C. Feng *et al.*, "Blockchain-based cross-domain authentication for intelligent 5G-enabled internet of drones," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 6224–6238, Apr. 2022.
[28] M. Tanveer, A. U. Khan, N. Kumar, and M. M. Hassan, "RAMP-IoD: A robust authenticated key management protocol for the internet of drones," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 1339–1353, Jan. 2022.
[29] B. LaMacchia, K. Lauter, and A. Mityagin, "Stronger security of authenticated key exchange," in *Provable Security*, W. Susilo, J. K. Liu, and Y. Mu, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, vol. 4784, pp. 1–16, series Title: Lecture Notes in Computer Science.
[30] Y. Li, M. Xu, and G. Xu, "Blockchain-based mutual authentication protocol without CA," *J. Supercomput.*, vol. 78, no. 15, pp. 17 261–17 283, Oct. 2022.
[31] S. Zou, Q. Cao, C. Wang, Z. Huang, and G. Xu, "A robust two-factor user authentication scheme-based ECC for smart home in IoT," *IEEE Sys. J.*, vol. 16, no. 3, pp. 4938–4949, Sep. 2022.
[32] C. Wang, D. Wang, Y. Tu, G. Xu, and H. Wang, "Understanding node capture attacks in user authentication schemes for wireless sensor networks," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 1, pp. 507–523, Jan. 2022.
[33] K. A. Farrea, Z. Baig, R. R. M. Doss, and D. Liu, "Provably secure optimal homomorphic signcryption for satellite-based internet of things," *Comput. Netw.*, vol. 250, p. 110516, Aug. 2024.
[34] I. Ullah *et al.*, "An efficient and secure multimessage and multireceiver signcryption scheme for edge-enabled internet of vehicles," *IEEE Internet Things J.*, vol. 9, no. 4, pp. 2688–2697, Feb. 2022.
[35] S. Zou *et al.*, "A physician's privacy-preserving authentication and key agreement protocol based on decentralized identity for medical data sharing in iomt," *IEEE Internet Things J.*, vol. 11, no. 17, pp. 29 174–29 189, Sep. 2024.
[36] P. Gope, O. Millwood, and N. Saxena, "A provably secure authentication scheme for RFID-enabled UAV applications," *Comput. Commun.*, vol. 166, pp. 19–25, Jan. 2021.
[37] M. Yahuza *et al.*, "An edge assisted secure lightweight authentication technique for safe communication on the internet of drones network," *IEEE Access*, vol. 9, pp. 31 420–31 440, 2021.
[38] B. Bera, A. K. Das, and A. K. Sutrala, "Private blockchain-based access control mechanism for unauthorized UAV detection and mitigation in internet of drones environment," *Comput. Commun.*, vol. 166, pp. 91–109, Jan. 2021.
[39] K. Huang, H. Hu, and C. Lin, "BAKAS-UAV: A secure blockchain-assisted authentication and key agreement scheme for unmanned aerial vehicles networks," *IEEE Internet Things J.*, vol. 11, no. 22, pp. 36 858–36 883, Nov. 2024.
[40] M. A. Khan *et al.*, "An efficient and provably secure certificateless blind signature scheme for flying ad-hoc network based on multi-access edge computing," *Electronics*, vol. 9, no. 1, p. 30, Jan. 2020, publisher: Multidisciplinary Digital Publishing Institute.
[41] ——, "An efficient and provably secure certificateless key-encapsulated signcryption scheme for flying ad-hoc network," *IEEE Access*, vol. 8, pp. 36 807–36 828, 2020.
[42] S. Yu and K. Park, "PUF-based robust and anonymous authentication and key establishment scheme for V2G networks," *IEEE Internet Things J.*, vol. 11, no. 9, pp. 15 450–15 464, May 2024.

Ruiyang Huang (Student Member, IEEE) is majoring in Cryptography Science and Technology at the School of Cyber Science and Engineering, Southeast University. He is a member of the endogenous security "Five-Color Stone Program" at Purple Mountain Laboratories (PML). His research interests lie at the intersection of cryptography and privacy-preserving technologies, security of unmanned swarm and multi-agent systems, and integrated sensing and communication. He is currently undertaking research internships focused on the security of multi-agent systems at FineLab at the Institute of Software at Peking University (ISPKU).

Ning Gao (Member, IEEE) received the Ph.D. degree in information and communications engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2019. From 2017 to 2018, he was a Visiting Ph.D. Student with the School of Computing and Communications, Lancaster University, Lancaster, U.K. From 2019 to 2022, he was a Research Fellow with the National Mobile Communications Research Laboratory, Southeast University, Nanjing, China. He is currently an Associate Professor with the School of Cyber Science and Engineering, Southeast University. His research interests include AI-enabled wireless communications and security, RIS and UAV communications.

Qiang Ni (Senior Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees from the Huazhong University of Science and Technology, China, all in engineering. He is currently a Professor and the Head of the Communication Systems Group, School of Computing and Communications, Lancaster University, Lancaster, U.K. His research interests include the area of future generation communications and networking, including green communications and networking, millimeter-wave wireless communications, cognitive radio network systems, non-orthogonal multiple access (NOMA), heterogeneous networks, 5G and 6G, SDN, cloud networks, energy harvesting, wireless information and power transfer, IoTs, cyber physical systems, AI and machine learning, big data analytics, and vehicular networks. He has authored or coauthored over 300 articles in these areas. He was an IEEE 802.11 Wireless Standard Working Group Voting Member and a contributor to the IEEE Wireless Standards.

Shi Jin (Fellow, IEEE) received the Ph.D. degree in communications and information systems from Southeast University in 2007. From June 2007 to October 2009, he was a Research Fellow with University College London, U.K., Adastral Park Research Campus. He is currently a Faculty Member of the National Mobile Communications Research Laboratory, Southeast University. His research interests include wireless communications, random matrix theory, and information theory. He and his co-authors have been awarded the 2011 IEEE Communications Society Stephen O. Rice Prize Paper Award in the field of communication theory, the IEEE Vehicular Technology Society 2023 Jack Neubauer Memorial Award, and the 2022 Best Paper Award and the 2010 Young Author Best Paper Award by the IEEE Signal Processing Society. He serves as an Area Editor for IEEE TRANSACTIONS ON COMMUNICATIONS and *IET Electronics Letters*. Previously, he was an Associate Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE COMMUNICATIONS LETTERS, and *IET Communications*.