Blockchain-Enabled Electrical Fault Inspection and Secure Transmission in 5G Smart Grids

Zhaolong Ning, *Senior Member, IEEE*, Handi Chen, *Graduate Student Member, IEEE*, Xiaojie Wang, *Member, IEEE*, Shupeng Wang, *Member, IEEE*, and Lei Guo

***Abstract*—The maturity of the 5th-Generation (5G) communication technology promotes a new round of industrial revolution and supports the high-quality development of economic society. However, owing to the scarce communication resources, costly labor and complex geographic environment, inspecting and maintaining electrical faults accurately and timely in a remote grid is rather challenging. To solve this problem, we comprehensively consider secure and efficient signal transmissions to construct an automatic grid fault inspection system and formulate a multi-objective optimization problem. Due to its complexity, we decompose it into two sub-problems, propose a blockchain-enabled secure transmission scheme (BEST) and an improved market matching (IMM) algorithm, correspondingly. Considering the latency magnitude difference between blockchain verification and intra-domain transmission, the BEST scheme integrates deep reinforcement learning-based improved proximal policy optimization training algorithm (DRIP) and A*-based bi-objective multi-destination optimization algorithm (ABOO) to achieve the intra-domain secure transmission. Based on the real city topology and the YouTube video service data statistics, our algorithms can optimize the network performance while guaranteeing the security of signal transmissions.**

***Index Terms*—Smart grids, blockchain, signal transmission, 5G communications.**

I. Introduction

SINCE the concept of the 5th-Generation (5G) communication technology was proposed in 2013, practical techniques, including Mobile Edge Computing (MEC) [1] and network resource virtualization, are developed to utilize heterogeneous resources efficiently [2]. The evolution of 5G-enhanced latency-sensitive networks, including medical [3], civil [4], industrial [5], military [6], has been promoted rapidly [7]. The coverage of smart grids is rapidly expanding, and the length of power lines is increasing at about 3% global annual growth rate [8]. As a latency-sensitive network, the challenges of electrical fault inspection and tough maintenance have always been rather difficult for maintainers. Power lines and electricity pylons are fault-prone due to climate change, human-made damage and so on, since they are always exposed to the wild. At present, the major inspection method is still manual inspection and maintenance [9]. However, the complex geographical environment and costly manual maintenance cost are the major obstacles to the accuracy and timeliness of manual inspection [10]. Therefore, the requirements of automated inspection paradigms are urgent to accurately detect and record the electrical fault.

However, according to Internet WorldStates and statistical bulletin of the communication industry by the ministry of industry and information technology in China, there are only 20% of land area, i.e., 6% of the earth surface, is covered by mobile communications. The communication resources are severely lacking, especially in remote areas without deployed celluar networks. Facing the challenge of 100% communication coverage expected by the 6th-Generation (6G) communication [11], leveraging Unmanned Aerial Vehicles (UAVs) with easy programmability and flexible mobility is a promising method to solve the challenge. The flexible networking capacities provide convenience for areas without network deployment and make up for traditional BS-based communication shortcomings.

A. Literature Review

The development of the 5G-enabled UAV signal transmission provides an opportunity to realize automatic fault inspection without the time and location limitations [12]. Margarita *et al.* in [13] viewed UAVs as air relay nodes to study 5G-enhanced dynamic and flexible air routing. Lin *et al.* in [14] studied autonomous UAV route for localizing ground objects. UAV without restricting by geographical location makes it suitable to cope with unforeseen natural disasters, such as earthquakes, hurricanes, and floods [15], [16]. In addition to natural disasters, UAV has been demonstrated its superiorities in public safety,

Manuscript received March 31, 2021; revised September 18, 2021; accepted October 13, 2021. Date of publication October 19, 2021; date of current version February 2, 2022. This work was supported in part by the National Natural Science Foundation of China under Grants 62025105, 61931019, 61971084, and 62001073, in part by Chongqing Talent Program under Grant CQYC2020058659, in part by the National Natural Science Foundation of Chongqing under Grant cstc2021ycjh-bgzxm0039 and Grant cstc2021jcyj-msxmX0031, in part by the Support Program for Overseas Students to Return to China for Entrepreneurship and Innovation under Grant cx2021003 and Grant cx2021053, and in part by the Dalian Young Science and Technology Star under Grant 2020RQ002. The guest editor coordinating the review of this manuscript and approving it for publication was Dr. Miaowen Wen. *(Corresponding authors: Xiaojie Wang; Shupeng Wang.)*

Zhaolong Ning is with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China, and also with the School of Software, Dalian University of Technology, Dalian 116024, China (e-mail: z.ning@ieee.org).

Xiaojie Wang and Lei Guo are with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: xiaojie.kara.wang@ieee.org; guolei@cqupt.edu.cn).

Handi Chen is with the School of Software, Dalian University of Technology, Dalian 116024, China (e-mail: hardychen@mail.dlut.edu.cn).

Shupeng Wang is with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100864, China (e-mail: wangshupeng@iie.ac.cn).

Digital Object Identifier 10.1109/JSTSP.2021.3120872

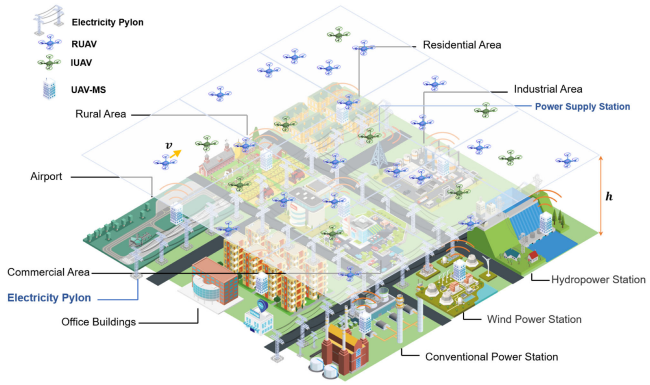


Fig. 1. An illustrative scenario for UAV-enhanced smart grids.

emergency scenario, intelligent transportation and industrial production [17], [18], gradually getting close to daily society.

Smart grid, as shown in Fig. 1, is a development priority in the industrial Internet of Things (IoTs). Leveraging UAVs with cameras and image recognition technique to inspect power lines and electricity pylons is a promising research direction. Samir *et al.* [19] jointly optimized the UAV route and ratio resource allocation to maximize the number of the served IoT devices in the city. Gapeyenko *et al.* [20] investigated the blocking issue of line-of-sight communications caused by urban buildings and proposed the line-of-sight probability model of UAV communication system based on Manhattan's Poisson line process. Shen *et al.* [21] proposed a data collection scheme to collect trust data of IoT devices while guaranteeing the credibility of data collection. There have been abundant investigations on the UAV applied in resourceful urban scenarios.

Generally, UAV investigations focus on the urban scenario with sufficient communication resources, and studies [19], [20] did not consider the security issue. For remote areas without network coverage, the fault signal needs to be transmitted to the areas within Power Supply Station (PSS) coverage by UAVs for uploading. Zhou *et al.* [10] studied the power line inspection by industrial Internet of UAVs in a smart grid to optimize energy efficiency. However, during long-distance transmissions, UAVs may be attacked at any time, which results in huge loss, such as power stealing, information leakage, and so on. Zhou *et al.* [22] proposed a secure UAV MEC system with a low-complexity iterative algorithm to maximize the minimum secrecy capacity. It sacrifices transmission performance to ensure security, which is not suitable for the latency-sensitive network. A monotonic optimization-based scheme was proposed in [23] to ensure secure transmissions. However, it is not suitable for UAV communications. In addition, some other literatures leveraged blockchain to guarantee security [24], which is a distributed ledger. This paper intends to design a 5G-enhanced intelligent electrical fault inspection system with secure transmissions in city-wide smart grids, including urban and remote areas.

B. Research Challenges

How to evaluate and quantify the security of blockchain verification, especially in remote areas, is rather challenging. Although this automatic fault inspection system has great potentials, there are still several challenges of establishing such a system:

- Grid attacks, such as the infamous Ukraine power grid attack event and the Venezuelan large-scale grid attack event, seriously threaten the grid security, resulting in huge losses from wasting resources to the leakage of secrets. Thus, how to efficiently transmit the fault signal with security and quantify the security are rather challenging.
- In the city grid, the electrical fault maintenance is a latency-sensitive issue. However, it is infeasible to schedule hundreds of UAVs simultaneously for long-distance signal transmissions due to the large-scale city topology and limited UAV power capacity. How to schedule UAVs with the optimized transmission route across large-scale city topology deserves for investigation.
- To optimize the performance and security of signal transmissions simultaneously, a multi-objective problem needs to be formulated, which is rather difficult to make a satisfying trade-off among conflict objectives simultaneously.

C. Contributions

To solve the challenges mentioned above, this paper leverages Consortium Blockchain (CB) technique to guarantee security and formulates a multi-objective optimization problem to make a trade-off between security and signal transmission. We decompose the formulated problem into two sub-problems and propose the corresponding schemes. The contributions of this work are summarized as follows:

- We construct an automatic electrical fault inspection system for a city-wide grid and formulate a multi-objective optimization problem to maximize the signal transmission utility and network security. The formulated problem is decomposed into two sub-problems, i.e., secure collaborative transmission and uploading sub-channel allocation.
- For the first sub-problem, we propose a blockchain-enabled secure transmission scheme (BEST), including blockchain verification and collaborative transmission route selection. Correspondingly, we propose a deep reinforcement learning-based improved proximal policy optimization training algorithm (DRIP) and an A*-based bi-objective multi-destination optimization algorithm (ABOO) to complete the two steps, respectively.
- For the second sub-problem, we propose an improved market matching (IMM) algorithm to optimize the uploading latency by considering the limited spectrum resources and time-sensitive power fault.
- The real-world topology in Xi'an city (China) and the online YouTube video services are utilized to simulate the network topology and data size of the processed video signal for demonstrating the effectiveness of our schemes.

The rest of this paper is structured as follows: we illustrate the system model in Section II; the problem is formulated and decomposed in Section III; in Section IV, we specify two schemes for solving the decomposed sub-problems, followed by

TABLE I
MAIN NOTATIONS

Notations	Descriptions
$\mathcal{G}$, $\mathcal{G}^d$	Topologies of the city and domain d
$\mathcal{D}$, D	The set of domains and the number of domains in the city, respectively
$\mathcal{E}$, $e\langle d, d'\rangle$	The adjacency relation set of domains and adjacency relation between domain d and domain d', respectively
$\mathcal{D}^d$, D^d	The set of UAVs and the number of UAVs in domain d, respectively
$\mathcal{E}^d$, $e^d\langle i, j\rangle$	The set of connection relationships among UAVs and the connection relationship between UAV i and UAV j in domain d, respectively
$\widetilde{\mathcal{D}}$, $\widetilde{D}$	The domain set and the number of domains within PSS coverage, respectively
h, v	The fixed height and fixed speed of UAV flight, respectively
$\mathcal{S}$	The PSS of the city
$\mathcal{C}^d$	The UAV-MS in domain d
$\check{P}$, $\check{q}$	The probability of UAV being attacked and the number of abnormal nodes (being attacked), respectively
$\mathbf{P}_x$	The probability of successful consensus verification
p_i	The transmission power of UAV i
B^{u2m}, B^{u2s}, B^{u2u}	The bandwidths of U2M, U2S and U2U channels, respectively
$p_{i,j}$	The received power between UAV i and UAV j
g	The constant power gain factor based on the amplifier and antenna
A_x	The AoI of fault x
s_x, s_x^{in}, s_x^{out}	The data size of fault x, the packed transaction and verification output with fault x, respectively
l^{record}, l^{proc}	The CPU cycles required by recording and packing, respectively
f_i, F	The CPU frequencies of UAV i and UAV-MS, respectively
$l^{sign}, l^{hash}, l^{rc}$	The CPU cycles required by signature verification, hash value calculation and result comparison, respectively
$D^\star$	The number of consensus nodes
ω	The number of transactions in a block

the experimental analysis in Section V; finally, we conclude our work in Section VI.

II. SYSTEM MODEL

In this section, we first present the system overview to detail the elements and the secure transmission mechanism. Then, we specify each module in the system model, including the security, communication, latency, and utility models. The major notations are summarized in Table I.

A. System Overview

The research scenario is constructed as a city-wide power grid with power nodes and power lines as shown in Fig. 1. The power grid can be divided into multiple domains with different power node densities according to the road network and urban functional areas. The power grid is denoted as an undirected graph $\mathcal{G} = (\mathcal{D}, \mathcal{E})$, where $\mathcal{D} = \{1, 2, \ldots, D\}$ and $\mathcal{E} = \{e\langle d, d'\rangle, d, d' \in \mathcal{D} \wedge d \neq d'\}$ indicate the set of domains and their adjacency relations, respectively. If domains d and d' are connected, $e\langle d, d'\rangle = 1$. To improve inspection efficiency and reduce manual inspection cost, UAV with a camera and image recognition technique can be used to inspect electrical fault, such as lightning stroke and bird damage, due to its flexibility and controllability.

As illustrated in Fig. 1, each domain contains a UAV Management Station (UAV-MS) to charge and schedule UAVs for electrical fault inspection and transmission. The UAV-MS scheduling contributes to quickly obtaining the optimal path based on global states and reducing the redundant cost for fault transmission. All UAVs fly at a fixed altitude h and speed v. UAVs can be divided into Inspection UAVs (IUAVs) and Relay UAVs (RUAVs) according to different functions. If an electrical fault is detected, it will be recorded as a video and transmitted to PSS for maintenance. Domain d containing D^d UAVs is defined as $\mathcal{G}^d = (\mathcal{D}^d, \mathcal{E}^d)$, where $\mathcal{D}^d$ and $\mathcal{E}^d$ respectively indicate the sets of transmission nodes and connection relationship among nodes, i.e., $\mathcal{D}^d = \{1, 2, \ldots, D^d\}$ and $\mathcal{E}^d = \{e^d\langle i, j\rangle, i, j \in \mathcal{D}^d \wedge i \neq j\}$. The PSS controls the city-wide power grid within all domains, denoted as $\mathcal{S}$. To ensure the security of 5G signal transmissions, the Registration Authority (RA) is deployed on the PSS to authorize and register nodes participating in the CB.

According to the division of domains, the system can be divided into blockchain verification and transmission as illustrated in Fig. 2. Since diverse reasons cause electrical faults, IUAV only records the inspected fault and transmits the record to PSS for manual validation, for decreasing the unnecessary labor costs caused by misjudgments. As shown in Fig. 2, the specific processes are detailed as follows:

1) IUAV inspects the electrical fault of power nodes or power lines, records and packs it as a fault signal;
2) The transmission request packed as a transaction, containing transaction header and fault entries, is generated and sent to UAV-MS for blockchain verification. UAV-MS audits the signature generated by the elliptic curve digital signature algorithm to verify the integrality of the received fault signal. After verification, UAV-MS signs its signature on the transaction, and the verified transaction is added into the transaction pool;
3) A block with transactions is generated via selecting from transaction pool and broadcasted by the UAV-MS to other consensus nodes (UAV-MSs) for blockchain verification;
4) After the block is verified, it is stored in the CB. The information of the corresponding fault can be looked up in the blockchain for ensuring security;
5) If the request is verified, UAV-MS selects RUAVs and destination to optimize the transmission route of the fault signal; otherwise, it returns the error message and refuses to transmit;
6) When the fault signal is sent to the next domain, the transmission request needs to be verified again. Steps 2)$\sim$5) are repeated until the fault signal is sent to the domain within PSS coverage;
7) The RUAV within PSS coverage uploads the fault signal to PSS based on the selection of sub-channel;
8) Through verifying fault signal manually in PSS, maintainers are dispatched to the fault location for maintaining. Meanwhile, PSS sends rewards to the corresponding UAV-MSs for awarding each participating UAV.

B. Security Model

CB technique is leveraged to guarantee the security and protect individual privacy of 5 G communications. PSS utilized as

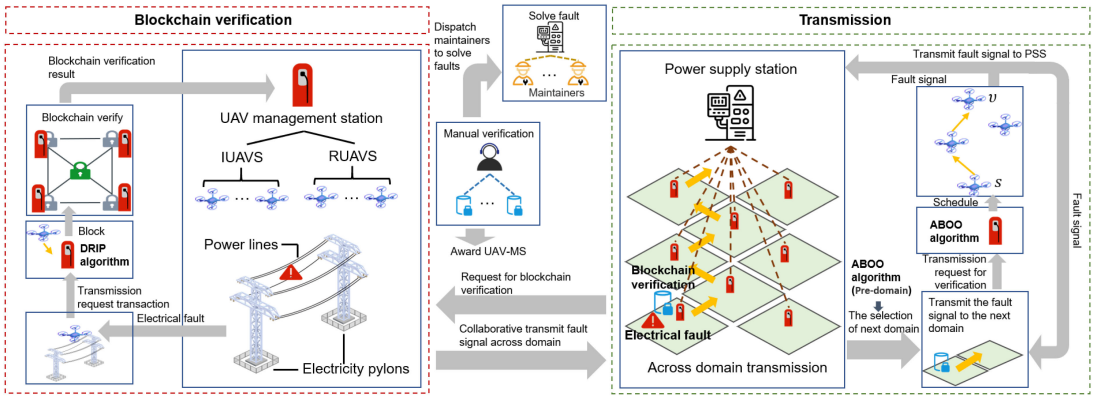


Fig. 2. An illustration of the detailed mechanism.

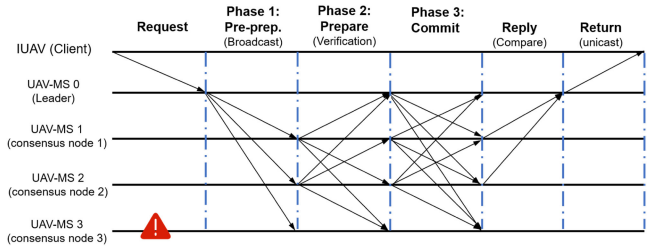


Fig. 3. Consensus processes for record transmission.

RA gives nodes distinct permissions through registration. UAV-MSs as consortium nodes are given consensus permission, and normal UAVs are only given read permission. Before generating a new block, blockchain verification with consensus protocol must be reached with the authorized blockchain consensus nodes. The process of the consensus protocol based on Practical Byzantine Fault Tolerance (PBFT) [25] is illustrated in Fig. 3, which can be divided into five stages.

1) UAV generates transmission request transaction, sends the transaction containing digital signatures and the hash value to the UAV-MS within the verification coverage. The UAV-MS, being activated by transaction transmission, is viewed as the leader in the consensus;
2) After receiving transactions, the hash value is calculated by a new block containing several transactions to construct the Merkel tree. The leader UAV-MS broadcasts the new block to other consensus nodes for verification. Consensus nodes can determine whether to accept the transaction;
3) After consensus nodes receive the new block, they first verify and audit the transaction integrity and legality. Each node integrates audit results in the digital signature of the received transaction and broadcasts to other nodes. Herein, the consensus nodes are regarded as semi-credible with probability $\check{P}$ of being attacked. The preparation phase is completed to commit until receiving messages from $2\check{q}$ distinct nodes, where $\check{q}$ indicates the number of abnormal nodes (i.e., consensus node 3 in Fig. 3) can be tolerated;
4) The committed results are broadcasted to other nodes for comparison. Once receiving more than $2\check{q}+1$ committed results, the request starts to be executed, and the generated block is written into the blockchain;
5) Finally, each consensus node forwards its committed results to the leader UAV-MS, and then the results are returned to the corresponding requester.

As the major operation to guarantee transmission security, the consensus mechanism evaluates the security of the blockchain. Considering consensus nodes may be attacked to become abnormal nodes, the number of consensus nodes needs to satisfy $D^{\star} \geq 3\check{q}+1$ [26]. Therefore, the probability of successful consensus verification can be represented as:

$$\mathbf{P}_x = 1 - \check{P}^{\frac{D^{\star}-1}{3}+1}, \tag{1}$$

which reflects the security of record transmission to protect individual privacy.

C. Communication Model

In our system, 5G-enhanced UAV communication involves three modes, i.e., UAV-to-Management station (U2M), UAV-to-UAV (U2U) and UAV-to-PSS (U2S), which are specified as follows:

1) U2M Communication Model: Each UAV is equipped with the Line-of-Sight (LoS) and Non-Line-of-Sight (NLoS) communication components. The free space pathloss can be computed by $\Lambda_i = 20\log f^{chan} + \log(4\pi/v^{light})$, where f^{chan} and v^{light} represent the channel frequency and the light speed, respectively. The pathloss of LoS and NLoS from UAV i to the UAV-MS $\mathcal{C}^d$ can be calculated by:

$$\Lambda_{i,d}^{los} = \Lambda_i + 20\log L_{i,d} + \lambda^{los}, \tag{2}$$

$$\Lambda_{i,d}^{nlos} = \Lambda_i + 20\log L_{i,d} + \lambda^{nlos}, \tag{3}$$

where λ^{los} and λ^{nlos} denote the average additional loss of LoS and NLoS, respectively. Variable $L_{i,d}$ indicates the distance between UAV i and UAV-MS $\mathcal{C}^d$, calculated by $\sqrt{(x_i-x_d)^2+(y_i-y_d)^2+h^2}$. The average pathloss can be obtained by $\Lambda_{i,d}^{avg} = \Lambda_{i,d}^{los}P_{i,d}^{los} + \Lambda_{i,d}^{nlos}P_{i,d}^{nlos}$, where $P_{i,d}^{los}$ indicates the probability of LoS communication, and $P_{i,d}^{nlos}$ indicates the probability of NLoS communication based on the antennas of UAV and the management station placed vertically.

Herein, $\Theta_{i,d}$ denotes the angle, and $P_{i,d}^{los} = 1/(1 + \dot{a}e^{-\ddot{a}\Theta_{i,d}-\dot{a}})$, $P_{i,d}^{nlos} = 1 - P_{i,d}^{los}$. Similar with [27], we also define S-curve parameters as coefficients $\dot{a}$ and $\ddot{a}$ to evaluate the probabilities of LoS communication. The average received power through the occupied channel can be obtained by:

$$p_{i,d}^{avg} = \frac{p_i}{10^{\Lambda_{i,d}^{avg}/10}}, \tag{4}$$

where p_i denotes the transmission power of UAV i. In U2M, the Orthogonal Frequency-Division Multiple Access (OFDMA) technique is leveraged to ensure the communication quality [28]. Therefore, the Signal to Noise Ratio (SNR) can be obtained by $\Gamma_{i,d} = p_{i,d}^{avg}/\sigma^2$, and the communication rate of U2M can be expressed by:

$$r_{i,d}^{u2m} = B^{u2m}\log_2(1 + \Gamma_{i,d}), \tag{5}$$

where B^{u2m} and σ^2 denote the UAV-MS sub-channel bandwidth and Additive White Gaussian Noise (AWGN), respectively.

2) U2S Communication Model: All fault signals need to be uploaded to PSS, which puts pressure on the limited spectrum resources. Therefore, the Non-Orthogonal Multiple Access (NOMA) technique is leveraged in U2S for sub-channel reusing [29]. The bandwidth of PSS is denoted as B^{u2s}, and the channel of $\mathcal{S}$ is divided into K^{u2s} sub-channels, which can be represented by $k \in \mathcal{K}^{u2s} = \{1, 2, \ldots, K^{u2s}\}$. The interference of sub-channel k can be calculated by:

$$I_{i^*,s,k} = \gamma_{x,i^*,k}\sum_{i'=1}^{\mathcal{I}_k}\gamma_{i,'s,k}p_{i,'s,k}^{avg}, \tag{6}$$

where $p_{i,'s,k}^{avg}$ can be obtained by (4), and $\gamma_{x,i^*,k}$ represents the binary variable of selecting channel k. The set of UAVs occupied sub-channel k is denoted as $\mathcal{I}_k$. Therefore, the Signal to Interference plus Noise Ratio (SINR) can be obtained by $\Gamma_{x,i^*,k} = p_{i,s,k}^{avg}/(\sigma^2 + I_{i,s,k})$. Correspondingly, the communication rate of U2S with the fixed PSS channel bandwidth B^{u2s} can be computed by:

$$r_{i,s,k}^{u2s} = \frac{B^{u2s}}{K}\log_2(1 + \Gamma_{x,i^*,k}). \tag{7}$$

3) U2U Communication Model: The free-space channel model is utilized to model U2U communications. The received power from UAVs i to j can be obtained by:

$$p_{i,j} = p_i g(L_{i,j})^{-\ddot{a}}, \tag{8}$$

where g denotes the constant power gain factor based on the amplifier and antenna. Variable $(L_{i,j})^{-\ddot{a}}$ represents the pathloss, and $L_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ with the fixed flight height h. In each time slot, only one U2U channel can be constructed to guarantee the communication quality. Therefore, the U2U communication rate for signal transmission with the fixed UAV channel bandwidth B^{u2u} is represented by:

$$r_{i,j}^{u2u} = B^{u2u}\log_2(1 + \frac{p_{i,j}}{\sigma^2}). \tag{9}$$

D. Latency Model

The latency of signal transmission affects the efficiency and timeliness of electrical fault maintenance. For clarifying, the latency can be simplified into four parts in our system, i.e., record latency, verification latency, transmission latency, and uploading latency. Herein, we introduce the concept of Age of Information (AoI) [30] to indicate the timeliness of record, and the AoI of fault x is calculated by $A_x = T_x^{rec} + T_x^{tran} + T_x^{up}$, where T_x^{rec}, T_x^{tran} and T_x^{up} indicate record latency, intra-domain transmission latency (including verification latency and transmission latency) and uploading latency, respectively.

1) Record Latency: The data size of fault information x recorded by IUAV i is denoted by s_x. The CPU cycles required by recording and processing a fault video are denoted as l^{record} and l^{proc}, respectively. Therefore, record latency T_x^{rec} of fault x inspected by i can be obtained by $(s_x l^{record} + s_x l^{proc})/f_i$, with computational capacity f_i of IUAV i.

2) Verification Latency: As illustrated in Fig. 3, the blockchain verification process mainly contains two steps: block generation and blockchain verification, i.e., $T_{x,d}^{bc} = T_{x,d}^{bg} + T_{x,d}^{bv}$. Herein, the block generation process can be specified as follows: 1) The transmission request packed as transaction is sent to UAV-MS for block generation, the transmission latency from UAV i to UAV-MS $\mathcal{C}^d$ can be denoted by $s_x^{in}/r_{i,d}^{u2m}$; 2) UAV-MS selects ω transactions from the transaction pool to verify signature and calculate hash value to construct Merkle tree, and the latency can be represented as $\omega l^{sign}/F_d + \sum_{h=0}^{\lceil\log_2\omega\rceil}\lceil\omega/2^h\rceil l^{hash}/F_d$. Herein, $\sum_{h=0}^{\lceil\log_2\omega\rceil}\lceil\omega/2^h\rceil$ indicates the number of Merkle tree nodes. Therefore, the block generation latency for fault signal x can be obtained by:

$$T_{x,d}^{bg} = \frac{s_x^{in}}{r_{i,d}^{u2m}} + T_{x,d}^{w} + 2\omega\frac{l^{sign}}{F_d} + \sum_{h=0}^{\lceil\log_2\omega\rceil}\lceil\frac{\omega}{2^h}\rceil\frac{l^{hash}}{F_d}. \tag{10}$$

After generating a block, the generator needs to validate the generated block based on consensus protocol. The verification process can be summarized as follows:

1) The block generator (UAV-MS) obtains the optimal number of consensus nodes and broadcasts the generated block. Similar to [31], we define ψ as the pre-defined parameter of broadcasting and comparison. The broadcasting latency among consensus nodes can be computed by $\psi D^\star \omega s_x^{in}$, where ωs_x^{in} and $D^\star$ indicate the block size and the number of consensus nodes, respectively. Obviously, the number of consensus nodes affects latency while guaranteeing security; The consensus latency can be obtained by $\omega(2\lceil\log_2\omega\rceil + 1)l^{hash}/F_d$;
2) After obtaining the consensus result, it is broadcasted to other UAV-MS for commitment. Similar to 1), the broadcasting latency can be obtained by $\psi D^\star s_x^{out}\omega$; The comparison latency of consensus results is computed by $s_x^{out}\omega l^{rc}/F_d$, where l^{rc} indicates the required CPU cycles for result comparison of one transaction;
3) The consensus result is unicasted to the block generator and added into the existing blockchain. The corresponding latency can be obtained by s_x^{out}/r^{d2d}.

Integrating the above steps, the latency of record x blockchain verification in domain d can be computed by:

$$T_{x,d}^{bv} = \psi D^{\star}\omega s_x^{in} + \omega(2\lceil \log_2 \omega \rceil + 1)\frac{l^{hash}}{F} + \psi D^{\star} s_x^{out}\omega + s_x^{out}\omega\frac{l^{rc}}{F} + \frac{s_x^{out}}{r^{d2d}} + T_d^w, \tag{11}$$

where T_d^w indicates the waiting latency during blockchain verification.

3) Transmission Latency: After fault signal verification, UAV-MS decides the signal transmission route, including the selections of RUAVs and the reachable destination based on the observed states. The transmission latency between UAVs i and j contains flight latency and communication latency, which can be computed by $L_{i,j}/v$ and $s_x/r_{i,j}^{u2u}$, respectively. We define binary variables $\alpha_{x,d}$ and $\beta_{x,d,i,j}$ to indicate the selection of domain d and the decision of signal transmission from UAVs i to j, respectively. If there is no available UAV at a certain moment, the fault signal can be transmitted to UAV-MS for temporary storage. Thus, the transmission latency in a domain can be calculated by $\sum_{i=1}^{D^d}\sum_{j=1}^{D^d}\beta_{x,d,i,j}(T_{x,d,i,j}^{f} + T_{x,d,i,j}^{comm})$, and the total transmission latency is denoted by:

$$T_x^{tran} = \sum_{d=1}^{D}\alpha_{x,d}[T_{x,d}^{bc} + \sum_{i=1}^{D^d}\sum_{j=1}^{D^d}\beta_{x,d,i,j}(T_{x,d,i,j}^{f} + T_{x,d,i,j}^{t})]. \tag{12}$$

4) Uploading Latency: When the fault signal is transmitted to the UAV-MS within PSS coverage, it will be uploaded to the PSS for manual verification. The uploading latency can be obtained by $\sum_{k=1}^{K}\gamma_{x,i^*,k}s_x/r_{i^*,s,k}^{u2s}$ based on (7). Therefore, the AoI of fault signal x can be represented by:

$$\begin{aligned} A_x = \frac{s_x l^{record} + s_x l^{proc}}{f_i^{\star}} + \sum_{d=1}^{D}\alpha_{x,d}\Bigg[\frac{s_x^{in}}{r_{i,d}^{u2m}} + 2\omega\frac{l^{sign}}{F} + \\ \sum_{h=0}^{\lceil \log_2 \omega \rceil}\lceil\frac{\omega}{2^h}\rceil\frac{l^{hash}}{F} + \psi D^{\star}\omega s_x^{in} + \omega(2\lceil\log_2\omega\rceil+1)\frac{l^{hash}}{F} \\ + \psi D^{\star}s_x^{out}\omega + s_x^{out}\omega\frac{l^{rc}}{F} + \frac{s_x^{out}}{r^{d2d}} + T_d^w + \\ \sum_{i=1}^{D^d}\sum_{j=1}^{D^d}\beta_{x,d,i,j}(T_{x,d,i,j}^{f} + T_{x,d,i,j}^{comm})\Bigg]. \\ + \sum_{k=1}^{K}\gamma_{x,i^*,k}\frac{s_x}{r_{i^*,s,k}^{u2s}}. \end{aligned} \tag{13}$$

E. Utility Model

The system utility is defined by the difference between the transmit reward and cost. After PSS receives the fault information, the reward of fault signal x is fed back to the corresponding UAV-MS.

1) Reward Model: The unit reward per bit is defined as ϕ^b, and the reward of fault signal transmission can be represented as $\phi^b s_x$. To stimulate UAVs collaboratively transmitting signals and reducing the impact of electrical faults, we leverage the sigmoid function to define the additional reward based on AoI. Thus, the reward related to electrical fault x is defined by:

$$\Phi(A_x, s_x) = \phi^b s_x(\frac{\varsigma}{1+e^{A_x}} + 1). \tag{14}$$

Herein, ς is set as the price coefficient to indicate the importance and reliability of the reported electrical fault. According to the characteristic of sigmoid function, $\varsigma/(1+e^{A_x}) \in [0, \varsigma/2]$.

2) Cost Model: Since UAVs are constrained by energy, we mainly focus on their costs. Generally, there are four working modes for the UAV, i.e., flight, hovering, computation and transmission.

Hovering and flight costs are obtained by the product of energy consumption and the unit price of energy. The required power in the hovering mode can be obtained by $p^h = W^2/\rho^{air}w^2v^{air}$ [32]. Herein, W and w denote the total weight and the width of the UAV, respectively. Variables ρ^{air} and v^{air} represent the air density and the airspeed, respectively. For the flight mode, the power is needed to overcome parasitic drag and lift the UAV [33]. Therefore, the power for the flight mode can be calculated by $p^f = \frac{1}{2}\kappa\mathscr{S}\rho^{air}v^{+3} + W^2/\rho^{air}w^2v^{+}$, where κ and $\mathscr{S}$ are the aerodynamic drag coefficient and the front-facing area, respectively. Variable v^{+} denotes the relative speed of the UAV through the air, i.e., $v^{+} = \|v + v^{air}\|$.

When a UAV performs transmission or computation operations, it is hovering in place. Thus, the cost of computation and transmission contains operation and hovering costs, which can be calculated by $s_x(l^{record} + l^{proc})\phi^c$ and $s_x\phi^t$, respectively. Thus, the total cost for transmitting electrical fault x becomes:

$$\begin{aligned} C_x = T_x^{rec}p^h\phi^e + s_x(l^{record} + l^{proc})\phi^c + \sum_{d=1}^{D}\alpha_{x,d}[T_{x,d}^{bc}p^h\phi^e \\ + \sum_{i=1}^{D^d}\sum_{j=1}^{D^d}\beta_{x,d,i,j}(T_{x,d,i,j}^{f}p^f\phi^e + T_{x,d,i,j}^{t}p^h\phi^e + \\ s_x\phi^t)] + \sum_{k=1}^{K}\gamma_{x,i^*,k}\frac{s_x p^h\phi^e}{r_{i,'s,k}^{u2s}} + s_x\phi^t. \end{aligned} \tag{15}$$

Thus, utility U_x^{total} of fault signal x can be obtained by:

$$U_x^{total} = \Phi(A_x, s_x) - C_x. \tag{16}$$

III. Problem Formulation

We formulate the secure signal transmission problem as a multi-objective optimization problem to maximize the total utility and security, i.e.,

$$P: \max_{D^{\star}}\mathbf{P}_x,$$

$$\max_{D^{\star},\omega,\alpha_{x,d},\{\beta_d\},\gamma_{x,i^*,k}} U_x^{total},$$

$$s.t.\, C1: P^{\frac{D^{\star}-1}{3}+1} \geq \mathbf{P}^{min}$$

$$C2: \alpha_d \in \{0,1\}, \forall d \in \mathcal{D}$$

$$C3: \beta_{x,d,i,j} \in \{0,1\}, \forall d \in \mathcal{D}, \forall i,j \in \mathcal{D}^d$$

$$C4: \gamma_{x,i^*,k} \in \{0,1\}, \sum_{k=1}^{K} \gamma_{x,i^*,k} = 1$$
$$C5: \gamma_{x,i^*,k}\Gamma_{x,i^*,k} \geq \Gamma^{min}$$
$$C6: D^\star \leq D$$
$$C7: 0 \leq \omega \leq \Omega$$
$$C8: \frac{\omega}{T_x} \geq \Theta^{min} \quad (17)$$

Herein, constraint $C1$ limits the minimum requirements for security. $C2$ and $C3$ restrict the decision variables of selecting transmission domains and RUAVs within a domain, respectively. $C4$ restricts the selection of U2S sub-channel, and $C5$ ensures that the minimum SINR should be satisfied. $C6$ restricts the upper bound number of consensus nodes in blockchain verification. The transaction decision in constraint $C7$ guarantees that the number of transactions ω should not exceed the transaction pool capacity, and $C8$ restricts ω to satisfy the minimum throughput.

Proposition 1: The formulated multi-objective optimization problem is NP-hard.

The proof of Proposition 1 can be found in Appendix A. The formulated multi-objective problem with distinct timescale variables is infeasible to be solved in polynomial time. Therefore, we decompose $P1$ into two sub-problems based on the split and integration method.

Proposition 2: The utility of fault signal x can be decomposed by maximizing the total reward and minimizing the total cost of domain d. The formulated multi-objective optimization problem P can be determined by four variables, i.e., security $\mathbf{P}_x$, signal transmission latency T_x^{tran}, signal transmission cost C_x^{tran} and uploading latency $T_{x,d'}^{up}$.

The proof of Proposition 2 can be found in Appendix B. According to Proposition 2, the formulated multi-objective joint optimization problem P can be divided into two sub-problems, i.e., $P1$ and $P2$. For problem $P1$:

$$P1: \max_{D^\star} \mathbf{P}_x,$$
$$\min_{D^\star,\omega,\alpha_{x,d},\{\beta_{x,d,i,j}\}} T_x^{tran},$$
$$\min_{D^\star,\omega,\alpha_{x,d},\{\beta_{x,d,i,j}\}} C_x^{tran},$$
$$s.t.\, C1, C2, C3, C6 - C8. \quad (18)$$

$P1$ intends to jointly optimize the cross-domain signal transmission performance while guaranteeing transmission security.

The purpose of $P2$ is selecting sub-channel to optimize the uploading process, i.e.,

$$P2: \min_{\gamma_{x,i^*,k}} T_x^{up},$$
$$s.t.\, C4 \text{ and } C5. \quad (19)$$

After solving $P1$, the record is transmitted to the domain within PSS coverage, and the RUAV needs to send it to PSS. Considering that the central PSS adjoins more than one domain, we set more than one RUAV in distinct domains to request for uploading electrical fault records to the PSS simultaneously. Thus, $P2^{(1)}$ is transformed to allocate sub-channels by minimizing the maximum uploading latency of all UAVs requesting for sub-channels, i.e.,

$$P2^{(1)}: \min_{\gamma_{u,k}} \max_{u \in \mathcal{U}^r} T_{u,k}^{up},$$
$$s.t.\, C4 \text{ and } C5, \quad (20)$$

where $\mathcal{U}^r$ indicates the set of RUAVs uploading record simultaneously.

IV. Our Solution

In this section, the BEST scheme is first designed to solve sub-problem $P1$. Specifically, we decompose it into two sub-problems to optimize the transmission security and cost, and propose the DRIP algorithm and the ABOO algorithm correspondingly. After that, an IMM algorithm solving problem $P2$ is detailed.

A. BEST Scheme for $P1$

Problem $P1$ is a joint optimization problem with distinct timescale variables, i.e., large-timescale optimization such as domain selection, small-timescale optimization such as consensus number, block size and relay selection. Since the variables in $P1$ are coupled, obtaining the optimal solution of different timescales in polynomial time is infeasible. In addition, the value of large-timescale variables is several orders of magnitudes higher than that of small-timescale optimization. Therefore, we propose a joint optimization scheme to solve $P1$.

1) Scheme Overview: Given the fault signal size s_x and city topology $\mathcal{G}$, the BEST scheme is illustrated in Algorithm 1. The large-timescale optimization variable $\{\alpha_{x,d}\}$ is dynamically determined based on the shortest route in the city topology. The proposed scheme first selects the destination from the large-timescale perspective based on the shortest route algorithm. The intra-domain collaborative transmission is jointly optimized to determine the optimal solution in the feasible solution domain set.

When UAV-MS $\mathcal{C}^d$ receives the transmission request record of fault signal x, $\alpha_{x,d} = 1$. The second objective of $P1$ can be rewritten by $T_{x,d}^{bc} + \sum_{i=1}^{D^d}\sum_{j=1}^{D^d} \beta_{x,d,i,j}(T_{x,d,i,j}^{comm} + T_{x,d,i,j}^{f})$, which can be decomposed into two parts, i.e., blockchain verification latency and signal transmission latency (including transmission latency and flight latency). The third objective of $P1$ can be transformed into $T_{x,d}^{bc} p^h \phi^e + \sum_{i=1}^{D^d}\sum_{j=1}^{D^d} \beta_{x,d,i,j}[(T_{x,d,i,j}^{comm} p^h + T_{x,d,i,j}^{f} p^f)\phi e + s_x \phi^t]$, including verification and transmission costs. Therefore, $P1$ can be optimized with two steps. For $P1^{(1)}$, the blockchain verification optimization can be transformed by maximizing the security of transmission, minimizing transmission latency and minimizing transmission cost. We can observe that the cost of verification is related to latency. Therefore, the objective of the

Algorithm 1: The Procedure of BEST Scheme.

The domain set $\widetilde{\mathcal{D}} = \{1, \cdots, \widetilde{D}\}$;
while *Fault signal* x *has not been transmitted to* $\widetilde{d} \in \widetilde{\mathcal{D}}$ **do**
  Optimize the numbers of consensus node $D^\star$ and transaction ω based on Algorithm 2;
  if x *is verified* **then**
    Construct transmission topology by Algorithm 3;
    Obtain destination decision $\alpha_{x,d'}$ and relay selection $\{\beta_{x,d,i,j}\}$ by Algorithm 4;
    Transmit x to the destination based on $\alpha_{x,d'}$ and $\{\beta_{x,d,i,j}\}$;
  else
    Record the error information;
    return error information.
return the AoI and cost of signal x.

first optimization step can be represented as:

$$P1^{(1)}: \max_{D^\star} \mathbf{P}_x,$$
$$\min_{D^\star,\omega} T^{bc}_{x,d},$$
$$s.t.\, C1, C6 - C8. \tag{21}$$

After fault signal x is verified, the transmission route of signal x can be optimized by solving the following sub-problem:

$$P1^{(2)}: \min_{\alpha_{x,d'},\{\beta_{x,d,i,j}\}} T^{tran}_{x,d},$$
$$\min_{\alpha_{x,d'},\{\beta_{x,d,i,j}\}} C^{tran}_{x,d},$$
$$s.t.\, C2 \text{ and } C3. \tag{22}$$

2) DRIP Algorithm for $P1^{(1)}$: The feasible solution set of problem $P1^{(1)}$ constitutes the Pareto optimal solution set [34]. The objective optimization problem is determined by two decision variables, i.e., the numbers of consensus node $D^\star$ and transaction ω. For simplicity, we set $Z_{x,d}$ to denote the risk cost, which is defined as the product of risk probability (i.e., $1 - \mathbf{P}_x$) and the latency, i.e.,

$$Z_{x,d} = (1 - \mathbf{P}_x)T^{bc}_{x,d} = (\check{P}^{\frac{D^\star - 1}{3}+1})(T^{bg}_{x,d} + T^{bv}_{x,d}), \tag{23}$$

where $T^{bg}_{x,d}$ and $T^{bv}_{x,d}$ are defined in equations (10) and (11), respectively. $Z_{x,d}$ reaches the minimum value only if both $(1 - \mathbf{P}_x)$ and $T^{bc}_{x,d}$ reach the minimum value. Therefore, problem $P1$ can be transformed into:

$$P1^{(1)'}: \min_{D^\star,\omega} Z_{x,d},$$
$$s.t.\, C1, C6 - C8. \tag{24}$$

The DRIP strategy training algorithm based on Proximal Policy Optimization (PPO) [35] algorithm, a novel policy gradient algorithm proposed by Schulman *et al.*, is designed to solve $P1^{(1)'}$ via optimizing the security of consensus. PPO proposes a new objective function that can be updated with small batches in multiple training steps, overcoming the obstacle that the step length is difficult to be determined in traditional policy gradient. The system state, action and reward function are detailed as follows:

System state: At time slot t, UAV-MS collects environment states, consisting of available computation capacity, UAV-MS communication rate, blockchain verification waiting latency and the combination strategy, i.e.,

$$\mathbb{S} = \{s_t | s_t = [F_d(t), r^{d2d}(t), T^w_{x,d}(t), D^\star(t), \omega(t)], t \in \mathbb{T}\}, \tag{25}$$

where $T^w_{x,d} \in \mathbf{T}^w$, $\mathbf{T}^w$ indicates the possible waiting latency set for generating block. Herein, the system states indicate the current dynamic blockchain environment to minimize the indicator.

System action: Considering optimizing two actions simultaneously, direct optimization of combination actions brings disaster to the action space dimension. To simplify, we supply the combination strategy as the state. The adjustment of the strategy is defined as actions to decrease the action space from $D \times \Omega$ to 3×3. The system actions are defined as follows:

$$\mathbb{A} = \{a_t | a_t = [a_{D^\star}(t), a_\omega(t)]\}, \tag{26}$$

where $a_{D^\star}(t)$ and $a_\omega(t)$ represent the adjustment of consensus nodes and transactions in time slot t, $a_{D^\star}(t)$ and $a_\omega(t) \in \{-1, 0, 1\}$. After iterations, the integrated strategies will converge to a strategy.

Reward function: Based on (24), the reward function obtained by UAV-MS can be constructed as:

$$\mathcal{R}(s_t, a_t, s_{t+1}) = -Z_{x,d}(s_t, a_t, s_{t+1}). \tag{27}$$

Starting from initial time slot, UAV-MS interacts with the environment following policy $\pi(a_t, s_t)$. Therefore, the accumulated reward can be obtained by:

$$\mathbf{R} = \sum_{t=0}^{\infty} \eta^t \mathcal{R}(s_t, a_t, s_{t+1}), s_t \in \mathcal{S}, \tag{28}$$

where η denotes the discount factor of reward. The optimization objective is finding the optimal policy $\pi^\star$ to maximize $\mathbf{R}$. We propose a Deep Reinforcement Learning (DRL)-based method to obtain the optimal policy. The pseudo-code of DRIP is specified in Algorithm 2.

At first, the iteration starts with a state, containing an initial combination strategy and an initial adjustment policy. Initial policy is executed to collect the observation. Similar to asynchronous advantage actor-critic algorithm, the generalized advantage estimator is calculated to make a compromise between variance and bias, i.e.,

$$\mathfrak{A}_t = \sum_{t'=0}^{\infty} \eta^{t'} [r_{t+t'} + \eta v(s_{t+t'+1}) - v(s_{t+t'})]. \tag{29}$$

The probability of each action in adjustment policy is updated for K epochs. In each epoch, the adjustment policy is updated based on the integration of loss functions $\mathbb{L}^{CLIP}_t(\theta)$ and $\mathbb{L}^V_t(\theta)$,

Algorithm 2: Pseudo-code of the DRIP Algorithm.

Initialize parameter θ randomly to obtain π_θ and the initial adjustment policy $\pi_{\theta_{old}}$ with $\theta_{old} \leftarrow \theta$;
for *Iteration* $i \in \{1, 2, \cdots, I\}$ **do**
  for $n \in \{1, 2, \cdots, N\}$ **do**
    Run policy π_θ for T time slots and collect $< s_t, a_t, r_t >$;
    Compute the generalized advantage estimator $\mathfrak{A}_t$, $t \in \{1, 2, ..., T\}$ according to (29) in each time step;
  for *epoch* $k \in \{1, 2, \cdots, K\}$ **do**
    Compute $\mathbb{L}^{ppo}(\theta)$ according to (30);
    Optimize policy with $\pi_{\theta_{old}} \leftarrow \pi_\theta$ according to $\mathbb{L}^{ppo}(\theta)$;
  Synchronise the policy with $\theta_{old} \leftarrow \theta$;

which is represented as:

$$\mathbb{L}^{ppo}(\theta) = \mathbb{E}_t[\mathbb{L}_t^{CLIP}(\theta) - \mu \mathbb{L}_t^V(\theta)], \tag{30}$$

where $\mathbb{L}_t^V(\theta)$ indicates the difference between estimation and reality, i.e., $\mathbb{E}_t[v'_{\pi_\theta}(s_t) - v(s_t)]$. Herein, $v'_{\pi_\theta}(s_t)$ and $v(s_t)$ denote the value-based network with parameter θ in state s_t and the real value function, respectively. $\mathbb{L}_t^{CLIP}(\theta)$ is defined to limit the rangeability of surrogate. It can be computed by $\mathbb{E}_t[\min(r_t(\theta)\mathfrak{A}_t, clip(r_t(\theta), 1-\varepsilon, 1+\varepsilon)\mathfrak{A}_t)]$, where $r_t(\theta)$ indicates the policy probability ratio, i.e., $r_t(\theta) = \pi_\theta(a_t|s_t)/\pi_{\theta_{old}}(a_t|s_t)$. The clip function is defined as $clip(r_t(\theta), 1-\varepsilon, 1+\varepsilon)\mathbb{A}_t$, indicating removing inventive for moving r_t outside of the interval $[1-\varepsilon, 1+\varepsilon]$ to modify the surrogate objective by clipping the probability ratio. After obtaining the updated parameters, the policy is updated.

3) Algorithm for $P1^{(2)}$: $P1^{(2)}$, formulated as a joint optimization problem, intends to minimize the transmission cost, involving both the large-timescale variable domain selection $\alpha_{x,d}$ and the small-timescale variable relay selection $\beta_{x,d,i,j}$. According to these two objectives in $P1^{(2)}$, the path between i and j can be represented as $e_{d,i,j} = (T_{i,j}, C_{i,j})$, where $T_{i,j}$ and $C_{i,j}$ indicate the latency and cost from the locations of UAVs i and j. Herein, if $j = \mathcal{C}^d$, it indicates there is no reachable RUAV for i, and the signal has been transmitted to $\mathcal{C}^d$ for temporary storage. If $i = \mathcal{C}^d$, it indicates the temporarily stored signal is sent to RUAV j for transmission. $P1^{(2)}$ intends to make a trade-off between latency and cost to obtain the Pareto-optimal route. To solve the problem, UAV-MS first extracts and constructs the intra-domain topology for route selection as illustrated in Algorithm 3. The UAV requesting for transmission is defined as the source node, which is represented as s, and the destination nodes are denoted as υ, $\upsilon \in \Upsilon^d$ where Υ^d denotes the set of destinations in domain d, and it can be obtained by Dijkstra algorithm from the current domain to PSS based on $\mathcal{G}$. We can observe the upper bound number of destination nodes is the number of adjacent domains. Algorithm 3 constructs a directed graph with a source node, multiple relay nodes (containing available RUAVs and UAV-MS) and destination nodes. Unreachable

Algorithm 3: Pseudo-code of Intra-domain Topology Extraction.

Input: Starting point s, available RUAVs U_d, UAV-MS C^d and city topoloty $\mathcal{G} = (\mathcal{D}, \mathcal{E})$
Output: Decisions $\{\beta_{x,d,i,j}\}$ and $\alpha_{x,d'}$ of the collaborative transmission of record x
1 $\Upsilon^d \leftarrow$ Dijkstra algorithm $(\mathcal{G})$;
2 Generate complete direct graph $\mathcal{G}^{d'} = (\mathcal{D}^{d'}, \mathcal{E}^{d'})$, $\mathcal{D}^{d'} = s \cup \mathcal{D}^d \cup \mathcal{C}^d \cup \Upsilon^d$;
3 **foreach** $e_{d,i,j} \in \mathcal{G}^{d'}, i \in \Upsilon_d$ *or* $j = s$ **do**
4 $\mathcal{G}^{d'}(\mathcal{D}^{d'}, \mathcal{E}^{d'}) \leftarrow \mathcal{G}^{d'}(\mathcal{D}^{d'}, \mathcal{E}^{d'}) - e_{d,i,j}$;
5 **foreach** $e_{d,i,j} \in \mathcal{G}^{d'}(\mathcal{D}^{d'}, \mathcal{E}^{d'})$, $i \in \mathcal{D}^{d'} \setminus \Upsilon_d \cup \mathcal{C}^d, j \in \mathcal{D}^{d'} \setminus s$ **do**
6 **if** $C_i - T_{i,j}^{tran} p^h \phi^e - \frac{L_{i,j}}{v} p^f \phi^e \leq C^{min}$ **then**
7 $\mathcal{G}^{d'}(\mathcal{D}^{d'}, \mathcal{E}^{d'}) \leftarrow \mathcal{G}^{d'}(\mathcal{D}^{d'}, \mathcal{E}^{d'}) - e_{d,i,j}$;
8 Obtain loop set $\mathfrak{L}_d \leftarrow$ *depth-first search*$(\mathcal{G}^{d'})$;
9 **while** $\mathfrak{L}_d \neq \varnothing$ **do**
10 **foreach** $l_d \in \mathfrak{L}_d$ **do**
11 Calculate in-degree and out-degree of each node;
12 **foreach** $e \in l_d$ **do**
13 **if** *the start node* i'_e *is not the source node of* $\mathcal{G}^{d'}$ *and has the maximum out-degree* **then**
14 Remove e starting from i'_e;
15 **if** *the end node* i''_e *has the maximum in-degree* **then**
16 Remove e ending at i''_e
17 **if** l_d *is broken* **then**
18 Remove l_d from $\mathfrak{L}_d$ and terminate the $foreach$ loop;
19 Obtain the extracted topology $\mathcal{G}^{d'}(\mathcal{D}^{d'}, \mathcal{E}^{d'})$;
20 Input $\mathcal{G}^{d''}(\mathcal{D}^{d'}, \mathcal{E}^{d''})$ to Algorithm 4 to obtain the optimal transmission route.

paths, i.e., the paths end of the source node and the edges start from the destination node, are regarded as redundant paths and deleted as shown in Lines $3 \sim 7$. Then, loops are removed from the graph to further extract intra-domain transmission topology $\mathcal{G}^{d'}$ [36] for obtaining a collaborative transmission route (Lines $8 \sim 18$).

The signal transmission topology constructed by Algorithm 3 is used as the input part for transmission route optimization with the ABOO algorithm (Algorithm 4). Algorithm 4 intends to find a Pareto-optimal route for UAV collaborative transmissions. An open list with all nodes $\mathcal{R}_o$ and a null close list $\mathcal{R}_c$ are initialized, which represent the sets of unvisited nodes and visited nodes, respectively. The route $\mathbb{R}$ from source node n_0 to node n_m is a set of order list with $\{n_0, n_1, \ldots, n_m\}$. The lexicographic order is defined to represent the relationship among paths. For vectors $e_1 = (T_1, C_1)$ and $e_2 = (T_2, C_2)$, if $T_1 < T_2$ or $T_1 = T_2 \wedge C_1 < C_2$, the relationship is defined as $e_1 \overset{o}{<} e_2$, i.e., e_1 is lexicographic smaller than e_2 in order. If

$T_1 \leq T_2 \vee C_1 \leq C_2$ is satisfied, the relationship is defined as path e_1 dominates e_2. The Pareto-optimal route $\mathbb{R}^{opt}$ cannot be dominated by any other path. For route $e \in \mathbb{R}_{i,j} \setminus \mathbb{R}_{i,j}^{opt}$, at least one path $e_o \in \mathbb{R}_{i,j}^{opt}$ satisfies e_o dominates e. The optimal route from source node s to destination node υ can be denoted as $\mathbb{R}_{s,\upsilon}$. Herein, in Pareto-optimal route set, the vector of the shortest path in latency can be denoted as $(\dot{T}_{i,j}, \widehat{C}_{i,j})$, and the vector of the shortest path in cost can be denoted as $(\widehat{T}_{i,j}, \dot{C}_{i,j})$. Routes in Pareto-optimal route set are ordered by lexicographic order.

The sub-path obtainment process leads to a fast search of the shortest path between source and destination nodes. $A^{\star}$ searching algorithm [37] is leveraged to obtain the Pareto-optimal path by label setting. During each iteration, obtaining the smallest path $e_{s,i} = (T_{s,i}, C_{s,i})$ among all available paths with respect to all nodes is selected. Then, the obtained path is extended to the next node j, i.e., $e_{s,j} = (e_{s,i}(T_{s,i}) + T_{i,j}, e_{s,i}(C_{s,i}) + C_{i,j})$.

Proposition 3: The cost C_x of fault signal x is positively correlated with AoI A_x.

The proof can be founded in Appendix C. According to Proposition 3, we can observe that latency dominates the optimal route selection. Therefore, the temporary sub-path is selected with the non-dominated one with the minimum latency, i.e., $\min(T_{s,i} + \dot{T}_{i,j})$. The dominated paths are removed from the obtained paths and added node j into $\mathcal{R}_c$. When all non-dominated paths to available destination nodes are obtained, the search terminates.

In a multi-objective problem, a criterion is needed for sub-path selection. We define a route key-value denoted as $\Delta(e_{s,i})$ to evaluate the route with the minimum extra latency of sub-path assignment for reaching the destination, which is computed by:

$$\Delta(e_{s,i}) = (T_{s,i} + \dot{T}_{i,j} - \dot{T}_{s,j}). \tag{31}$$

The maximum extra latency can be denoted as $\widehat{\Delta}_{s,\upsilon} = \widehat{T}_{s,\upsilon} - \dot{T}_{s,\upsilon}$ which indicates the latency difference with two distinct objectives. During searching, the sub-path is selected based on the non-decreasing extra latency, i.e., $T_{s,\upsilon} - \dot{T}_{s,\upsilon}$. When Pareto-optimal routes to destination nodes are found, i.e., the minimum key value of the sub-path in the residual network is larger than $\max_{1 \leq j \leq \upsilon} \widehat{\Delta}_{s,j}$, the algorithm terminates.

Proposition 4: All Pareto-optimal routes from source node s to destination node $\upsilon \in \Upsilon_d$ can be obtained once the algorithm terminates.

The proof of Proposition 4 is specified in Appendix D.

B. Algorithm for $P2$

After $P1$ is solved, the transmission cost and latency can be obtained. The electrical fault signal should be uploaded to the PSS, i.e., sub-channel allocation problem $P2^{(1)}$ needs to be solved. Considering the city-wide grid is managed by a PSS, multiple UAVs may request for signal transmissions simultaneously. To maintain the sub-channel matching process, we propose an IMM algorithm, which can resolve the unilateral dominance problem caused by the matching order in traditional matching algorithms. The pseudo-code is specified in Algorithm 5 and Algorithm 6, which are specified in Appendix E.

Algorithm 4: Pseudo-code of Generating Pareto-optimal Routes.

Input: $\mathcal{G}^{d''}(\mathcal{D}^{d'}, \mathcal{E}^{d''})$, source node s, destination node Υ_d
Output: $\{\beta_{x,d,i,j}\}$ and $\alpha_{x,d}$
Initial $e_s = (0, 0)$;
Initial temporary list $\mathcal{R}$, open list $\mathcal{R}_o$ close list $\mathcal{R}_c$;
Add e_s into $e_{s,\upsilon}$ ($\upsilon \in \Upsilon_d$) and temporary list $\mathcal{R}$;
Obtain $\dot{T}_{s,i}$, $\widehat{C}_{s,i}$, $\widehat{T}_{s,i}$ and $\dot{C}_{s,i}$ for $i \in \mathcal{D}_d{}'$;
foreach $\upsilon \in \Upsilon_d$ **do**
    Obtain $\dot{T}_{i,\upsilon}$, $\widehat{C}_{i,\upsilon}$, $\widehat{T}_{i,\upsilon}$ and $\dot{C}_{i,\upsilon}$ for $i \in \mathcal{D}_d{}'$ with backward Dijkstra algorithm;
    Compute $\widehat{\Delta}_{\upsilon} = \widehat{T}_{s,\upsilon} - \dot{T}_{s,\upsilon}$;
while $\min_{i \in \mathcal{D}_d{}'} \Delta(e) \leq \max_{i \in \mathcal{D}_d{}'} \widehat{\Delta}_i$, *for* $\upsilon \in \Upsilon$, $e \in \mathcal{R}_o$ **do**
    Extract e with $\min_{i \in \mathcal{D}_d{}'} \Delta(e)$;
    foreach $e_{d,i,j} \in \mathcal{E}_d{}''$ **do**
        $e_{s,j} = (e_{s,i}(T_{s,i}) + T_{i,j}, e_{s,i}(C_{s,i}) + C_{i,j})$;
        if $e_{s,j}$ *is not dominated by any path* **then**
            $\beta_{x,d,i,j} = 1$;
            Add $e_{s,j}$ to route e;
        if $e_{s,j}$ *dominates* $e \in \mathcal{R}$ **then**
            Remove e from $\mathcal{R}$ and $e_{s,j}$;
            Add e into $\mathcal{R}_c$;
            $\beta_{x,d,i,j} = 0$;
Obtain Pareto optimal route $\{e_{s,\upsilon}\}$ for destination Υ_d with $\{\beta_{x,d,i,j}\}$;
Extract the non-dominated optimal route $e_{s,\upsilon'}$ and variable $\{\beta_{x,d,i,j}\}$;
Destination node υ' corresponds to domain d', $\alpha_{x,d'} = 1$.

As illustrated in Algorithm 5, the sub-channel allocation issue involves two kinds of entities, i.e., UAVs and sub-channels. These entities are respectively denoted by $\mathcal{U}^r = \{1, \ldots, U^r\}$ and $\mathcal{K} = \{1, \ldots, K\}$. Each UAV maintains a preference list ordered by the sub-channel states, i.e., the number of users occupying the sub-channel. Entities with the worst state in the UAV's preference list are considered as the worst matching entity. Since each UAV can only occupy one sub-channel, the UAV's matching capacity is set to one, i.e., $C4$ in (17). We define the transmission latency as an urgent value to evaluate UAVs. Correspondingly, the preference list of each sub-channel is ordered by the priority of UAVs. $C5$ in (17) defines the worst case of sub-channel selection. Obtaining stable matching results requires no blocking pairs exist in the matching list.

The procedure starts with list $\mathbb{M}$ containing all entities (contain UAVs and sub-channels) and empty temporary queue $\mathbb{M}_T$. UAV entities and sub-channels take turns to send matching requests to others in the order of the preference list. Requesting entity compares the matching list with the previous results to replace the worse one, and the worse one is put into the temporary queue $\mathbb{M}_T$. It will be matched first in the next round until obtaining the optimal matching solution.

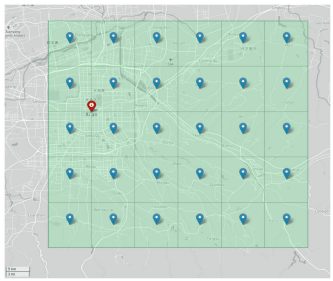


Fig. 4. The simulation topology map.

Proposition 5: The time complexity of Algorithm 5 is $\mathcal{O}(M^2)$, where $M = K + U^r$.

The proof of Proposition 5 can be found in Appendix F.

V. Experimental Analysis

In this section, the effectiveness of our proposed algorithms is demonstrated. We first introduce the simulation setup, and then present the evaluation and analysis of the experimental results.

A. Simulation Setup

These experiments are realized in a 64 b Windows 10 operating system computer, which is deployed with a 32.0 GB RAM, an Intel(R) Core(TM) i7-10700F CPU @ with 2.90 GHz frequency and an NVIDIA GeForce GT 1030. We utilize PyCharm 2020.02 version based on Python 3.8 to perform experiments.

Similar to [38], the grid is divided into square domains with 10 km based on the map of Xi'an city (China), as shown in Fig. 4. As an important node of the Belt and Road policy, this city has dense power nodes in prosperous urban areas. In addition, the remote mountainous areas are average about 25 km away from the city center, satisfying our requirements of the investigated city includes both urban and remote areas. Herein, the coordinate of PSS is [34.27145,108.94243], which is set in the city center and marked with a red flag in Fig. 4. The number of sub-channels in U2S communication is set as 10, where one sub-channel can only be multiplexed by three devices due to NOMA constraints [39]. UAV-MS is set in the center of each domain, which is marked with a blue flag. The experimental area is set as 50 km $\times$ 60 km with latitude from 34.04662 to 34.49629 and longitude from 108.79694 to 109.37892, containing urban and remote mountainous areas. Since no dataset is available for electrical fault verification, we utilize the online YouTube video services in [40] to simulate the fault signal processed from the original video signal. According to the statistics in [40], the distribution of YouTube video is mainly concentrated in (0,100] MB, which is defined as the size interval of fault signal. The major constant parameters are specified in Table II.

For sub-problem $P1$, considering the distinct latency magnitude between blockchain verification and intra-domain transmission, the performance of the proposed BEST scheme is evaluated via analyzing the performance of $P1^{(1)}$ and $P1^{(2)}$, respectively. We first demonstrate the effectiveness of the DRIP algorithm. The comparison algorithms are selected:

TABLE II
Constant Parameters for Experiments

Parameter	Value
UAV flight height h and speed v	50 m and 10 m/s
AWGN σ^2	-96 dBm
UAV i transmission power p_i	23 dBm
Channel frequency f_c	1444 Hz
Channel bandwidth of U2U (B_u), U2M (B_m) and U2S (B_s)	2 MHz, 5 MHz and 20 MHz
CPU frequency of UAV (f_i) and UAV-MS (F)	1 GHz and 5 GHz
U2U power gains factor g	-31.5 dB
Collision avoidance communication distance between UAVs $L_{i,j}$	2 m
Required CPU cycles per bit for recording and packing electrical fault l^{record}	10^5 CPU cycles per bit
Required CPU cycles for verifying a signature l^{sign}, calculating a hash value l^{hash} and consensus results comparison l^{rc}	1.5×10^5 CPU cycles, 2×10^3 CPU cycles and 10^4 CPU cycles
Data size of transaction s^{in}	1 kb
U2X channel parameters λ^{los} and λ^{nlos}	1 and 20
U2X channel coefficients $\dot{a}$ and $\ddot{a}$	12 and 0.135
U2U pathloss coefficient c	2
Minimum blockchain verification security $\mathbf{P}^{min}$	0.96
Minimum blockchain SINR $\gamma_{x,i^*,k}$	0.1
Unit price per kilowatt-hour ϕ^e	1
Unit computation price per cycle ϕ^c	10^{-5}
Unit transmission price per bit ϕ^t	10^{-4}

- DRL-based Performance Optimization Framework (DPOF) [41]: It designs a modulable blockchain, optimizes block size and block interval based on DRL algorithm to maximize the throughput by considering the blockchain latency and security.
- Greedy in latency policy: It solves $P1^{(1)}$ by minimizing the block verification latency with security constraint.
- Greedy in security policy: Contrary to greedy in latency policy, it solves $P1^{(1)}$ by maximizing the successful verification probability with latency constraint.
- Random policy: It randomly selects strategies satisfying the constraints to solve $P1^{(1)}$. Once a feasible strategy is found, it is utilized without searching for better strategies.

To solve the bi-objective sub-problem $P1^{(2)}$, the proposed ABOO makes a trade-off between latency and cost. Considering the influence of UAV distribution, we respectively evaluate in random and standard distributions to simulate high-workload urban areas and low-workload remote areas, with three indicators including latency, cost and throughput. Four representative policies are leveraged:

- Joint Trajectory Design and Task Scheduling algorithm (TDTS) [17]: It integrates an average throughput maximization-based auction algorithm to determine the trajectory of UAVs based on the auction bidding obtained by latency approximation algorithm.
- Floyd in latency policy: It leverages the Floyd algorithm [42] to minimize the transmission latency without considering the transmission cost.
- Floyd in cost policy: Similar to Floyd in latency policy, it finds the shortest route in cost without considering the transmission latency.

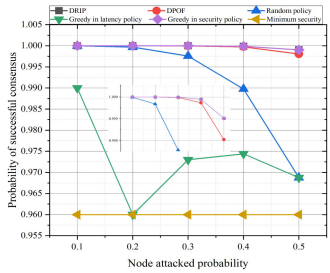
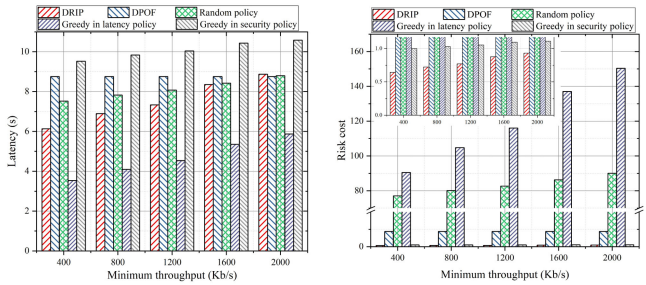


Fig. 5. Probability of successful consensus.



(a) Latency. (b) Risk cost.

Fig. 6. Performance with distinct throughput limitations.

- Random policy: It randomly selects sub-paths until the fault signal is transmitted to a reachable destination. Once the route is found, UAV-MS stops searching.

For sub-problem $P2$, the optimization objective is solved by the proposed IMM algorithm. Three policies are leveraged for comparison:

- Two-side matching algorithm [43]: Similar to our proposed IMM algorithm, it schedules the unlicensed spectrum by the constructed preference list and the priority list of users until all requesters are matched with sub-channels.
- Greedy policy: It traverses requesters in an order of priority. Each requester requests for sub-channel with the best state, which is also found by traversing.
- Random policy: It randomly selects sub-channel for requesters until all requesters are matched, and there is no unmatched conflicting pair.

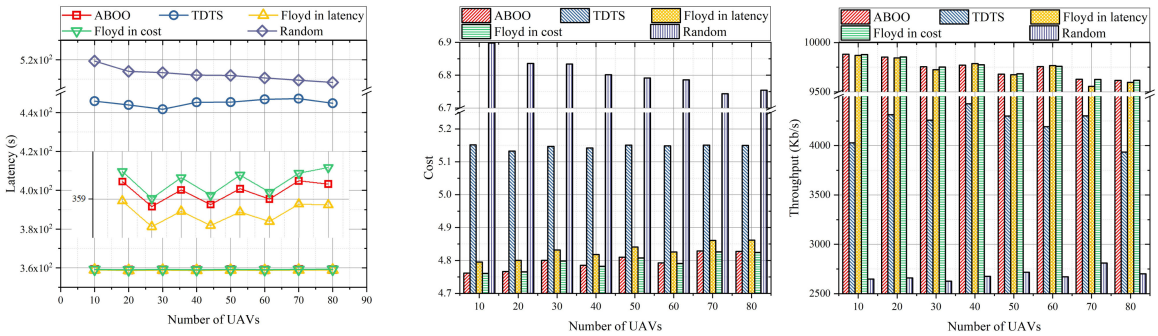
B. Performance Results

In this subsection, we evaluate and analyze the performance results of the proposed three schemes.

1) Performance of $P1$: Fig. 5 illustrates the probabilities of successful consensus by distinct policies with different being attacked probabilities under the fixed minimum security probability 0.96. We can observe that our proposed DRIP algorithm and DPOF algorithm are closest to the greedy in security policy with less than 0.3 attacked probability. However, the security of DPOF decreases gradually as the attacked probability becomes large, while DRIP is still closest to the optimal security, as shown in the enlarged image of Fig. 5. The security of random policy has a good performance when the probability of being attacked is low, while it begins to drop sharply when the being attacked probability increases to 0.3. In addition, it is noted that the greedy policy in latency does not have a fixed trend since it only intends to minimize the latency. Once the blockchain verification parameters satisfy the minimum security threshold, i.e., $\mathbf{P}_x \geq 0.96$, the device stops searching for the strategies with better security. That is because the performance of greedy in latency policy is only related to the being attacked probability and the minimum security limitation. The proposed DRIP algorithm, optimizing the risk cost with cost and security simultaneously, adjusts the blockchain verification parameters to make a trade-off between the two objectives of cost and security. The security is always guaranteed at a high level with distinct node attacked probabilities.

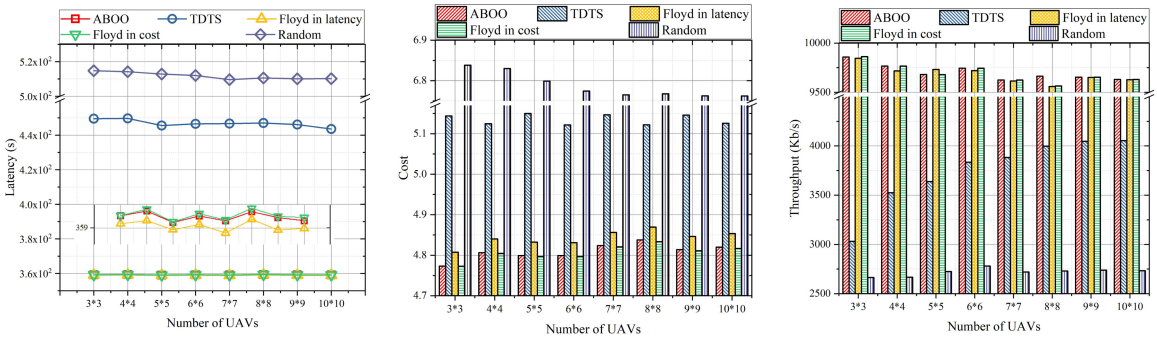
Fig. 6 shows the performance of latency and risk cost with different given throughput limitations. As shown in Fig. 6(a), we can observe that the latency of four policies (except DPOF) rises when the throughput limitation increases. Herein, the latency performance of our DRIP is second only to the greedy in latency policy. Integrating Fig. 6(b), DRIP produces the minimum risk cost. These results validate that the proposed DRIP makes a satisfied trade-off between security and latency, sacrificing acceptable latency for higher security. Compared with DPOF, DRIP sacrifices less latency with nearly 10 times less risk cost than DPOF. Integrating Fig. 5, these results demonstrate that optimizing risk cost integrating security and cost enables blockchain to perform better than traditional maximizing throughput. Herein, although the latency performance of greedy in latency policy and random policy is relatively better than other policies in Fig. 6(a), their risk costs in Fig. 6(b) illustrate the lack of security, which can also be observed in Fig. 5.

Fig. 7 illustrates the relationships between the number of UAVs and transmission performance in random distributions simulating urban areas. Herein, Floyd in latency and Floyd in cost policies are viewed as two benchmarks to indicate the optimal route with distinct objectives. It can be observed that the ABOO performance is between that of these two benchmarks. These results demonstrate that our ABOO, making a trade-off between transmission cost and latency, is more suitable for complex UAV topologies as shown in Figs. 7(a) and 7(b). It is noted that Floyd in latency and Floyd in cost policies have the same trend, which demonstrates the positive correlation between latency and cost in Proposition 3. Herein, the latency and cost of ABOO are averagely 20% and 7% lower than that of TDTS, respectively. That is because that the auction process of TDTS for signal transmission without global states is difficult to obtain a global optimal transmission route. In Fig. 7(c), the throughput of ABOO is averagely 130% higher than that of TDTS, indicating ABOO algorithm obtains the transmission route with better communication states from global perspective. Intuitively, the increase of UAVs provides more available routes for signal transmissions, as shown in random policy and TDTS. These results indicate that as the number of UAVs increases, the total transmission performance gradually stabilizes within a range. That is because collaborative transmission among UAVs brings extra latency and cost during transmission among UAVs. The proposed ABOO comprehensively considers route



(a) Latency. (b) Cost. (c) Throughput.

Fig. 7. Performance with distinct numbers of UAVs in random distributions.

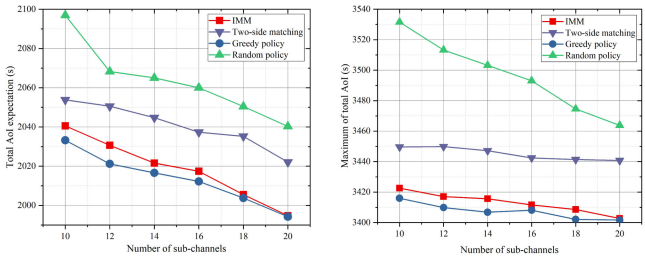


(a) Latency. (b) Cost. (c) Throughput.

Fig. 8. Performance with distinct numbers of UAVs in standard distributions.

selection and communication states, obtaining better performance in experiments than other schemes.

Fig. 8 shows the relationships between the number of UAVs and transmission performance in a standard distribution, simulating remote areas. Different from the performance of UAVs in the random distribution, each UAV has its fixed position, which greatly reduces the impact of the distance among available UAVs on performance during transmissions. We can observe that given a fixed UAV topology, the transmission performance is only related to the power state, communication state and the uniform deployment distance. These results indicate ABOO algorithm makes a satisfied trade-off between two optimal routes with distinct objectives compared with other schemes. In the case of random distribution, a larger UAV topology provides more available routes. However, for the standard distribution with a fixed UAV topology, the impact of the reachable state of the UAV on signal transmission is less than that of the flight distance generated by the topology. Compared with random distribution, the performance of signal transmission is more stable in the standard distribution as shown in Figs. 8(a) and 8(b). In Fig. 8(c), the throughput of the ABOO algorithm is averagely 150% higher than TDTS. That indicates UAV-MS has a bigger advantage for RUAVs based on TDTS when the experimental topology size is less than 8×8. That is because RUAVs based on a distributed auction scheme are inclined to greedily select the more efficient UAV-MS as the next relay based on TDTS, accumulating the communication load to decrease the throughput.

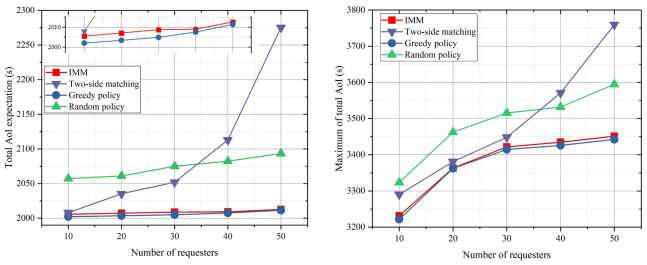


(a) Total AoI expectation. (b) Maximum of total AoI.

Fig. 9. Performance with distinct numbers of PSS sub-channels.

Until the topology size is larger than 8×8, the advantages of UAV utilized as relay based on TDTS are gradually obvious, and the throughput tends to be stable. The proposed ABOO, considering global optimal route and communication states, can obtain relatively high and stable throughput with distinct experimental topologies.

2) Performance for P2: Fig. 9 evaluates the total AoI expectation and the maximum total AoI with different numbers of available sub-channels. Total AoI is the sum of transmission AoI and uploading AoI, where the former is obtained by the ABOO algorithm to control route. We can observe that as the number of available sub-channels increases, the total AoI and the total AoI expectation both show a downward trends. That is because as the



(a) Total AoI expectation. (b) Maximum of total AoI.

Fig. 10. Performance with distinct numbers of requesters.

number of available spectrum resources increases, the interference impact caused by sub-channel reuse can be significantly improved. The total AoI expectation difference between the IMM algorithm and greedy policy is 7.34 s with 10 sub-channels and decreases to 0.53 s with 20 sub-channels. That illustrates that our algorithm performs better with multiple sub-channels. Similarly, the maximum total AoI difference is 27.01 s with 10 sub-channels and decreases to 1.03 s with 20 sub-channels. In Fig. 9(a), the performance of the IMM algorithm is closest to that of the greedy policy, which finds the optimal sub-channel allocation schemes through traversing. As the number of sub-channels increases, the performance of the IMM algorithm tends to be better, gradually. These experimental results demonstrate our IMM algorithm, updating the preference list and the worst case in the matched list at each iteration, obtains better allocation results and reduces the impact of unilateral dominance during matching compared with the two-side matching algorithm.

Fig. 10 shows the relationships between the total AoI expectation and the maximum total AoI with different numbers of requesters. In Fig. 10(a), we can observe that the two-side matching algorithm performs worse when the number of requesters is more than 30, because the obsolete preference list leads to matching conflicts. The pair with low priority is unmatched, which further causes the AoI to increase sharply. The proposed IMM constantly maintains and updates the preference lists of requesters and sub-channels at each iteration. Even if the requested amount of the sub-channel is large, it can also be closed to the optimal scheme, which confirms that the proposed IMM can adapt to the complex scenario with a large number of requesters.

VI. Conclusion

This paper has integrated the CB technique to propose an automatic secure and efficient electrical fault inspection system. The flexible networking characteristics of UAV communications and the security of CB for protecting privacy are leveraged to construct the system, considering the scarce 5 G communication resources of remote areas. The problem has been formulated as a multi-objective optimization problem, i.e., maximizing security and utility. To solve this problem, we have divided it into two sub-problems of signal transmission and signal uploading. Considering the complexity and cost magnitude difference for the first sub-problem, we have proposed the BEST scheme to transform it into a two-step optimization problem. The DRIP and the ABOO algorithms have been proposed to solve it. The IMM algorithm has been proposed to allocate the limited NOMA sub-channel resources by maintaining the priority and preference lists of users and sub-channels. Based on the real city topology of Xi'an city (China) and YouTube video services, the experiments have demonstrated the superiorities of our proposed schemes.

References

[1] Y. Wu, L. P. Qian, K. Ni, C. Zhang, and X. Shen, "Delay-minimization nonorthogonal multiple access enabled multi-user mobile edge computation offloading," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 3, pp. 392–407, Jun. 2019.

[2] G. Faraci, C. Grasso, and G. Schembra, "Design of a 5G network slice extension with MEC UAVs managed with reinforcement learning," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 10, pp. 2356–2371, Oct. 2020.

[3] Y. Jiang, Y. Ma, J. Liu, L. Hu, M. Chen, and I. Huma, "MER-WearNet: Medical-emergency response wearable networking powered by UAV-assisted computing offloading and WPT," *IEEE Trans. Netw. Sci. Eng.*, early access, Mar. 17, 2021, doi: 10.1109/TNSE.2021.3066598.

[4] S. Hayat, E. Yanmaz, and R. Muzaffar, "Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint," *IEEE Commun. Surv. Tut.*, vol. 18, no. 4, pp. 2624–2661, Oct.–Dec. 2016.

[5] H. Liu, Q. Chen, N. Pan, Y. Sun, Y. An, and D. Pan, "UAV stocktaking task-planning for industrial warehouses based on improved hybrid differential evolution algorithm," *IEEE Trans. Ind. Inform.*, vol. 18, no. 1, pp. 582–591, Jan. 2022.

[6] F. Xiong, A. Li, H. Wang, and L. Tang, "An SDN-MQTT based communication system for battlefield UAV swarms," *IEEE Commun. Mag.*, vol. 57, no. 8, pp. 41–47, Aug. 2019.

[7] J. Choi, "Single-carrier index modulation for IoT uplink," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 6, pp. 1237–1248, Oct. 2019.

[8] A. Golshani, W. Sun, Q. Zhou, Q. P. Zheng, and J. Tong, "Two-stage adaptive restoration decision support system for a self-healing power grid," *IEEE Trans. Ind. Inform.*, vol. 13, no. 6, pp. 2802–2812, Dec. 2017.

[9] M. A. Ali, Y. Zeng, and A. Jamalipour, "Software-defined coexisting UAV and WiFi: Delay-oriented traffic offloading and UAV placement," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 6, pp. 988–998, Jun. 2020.

[10] Z. Zhou, C. Zhang, C. Xu, F. Xiong, Y. Zhang, and T. Umer, "Energy-efficient industrial internet of UAVs for power line inspection in smart grid," *IEEE Trans. Ind. Inform.*, vol. 14, no. 6, pp. 2705–2714, Jun. 2018.

[11] X. You *et al.*, "Towards 6G wireless communication networks: Vision, enabling technologies, and new paradigm shifts," *Sci. China Inf. Sci.*, vol. 64, no. 1, pp. 1–74, 2021.

[12] H. Zhang, J. Zhang, and K. Long, "Energy efficiency optimization for NOMA UAV network with imperfect CSI," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 12, pp. 2798–2809, Dec. 2020.

[13] M. Gapeyenko, V. Petrov, D. Moltchanov, S. Andreev, N. Himayat, and Y. Koucheryavy, "Flexible and reliable UAV-assisted backhaul operation in 5G mmWave cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2486–2496, Nov. 2018.

[14] C. Lin, G. Han, X. Qi, J. Du, T. Xu, and M. Martínez-García, "Energy-optimal data collection for unmanned aerial vehicle-aided industrial wireless sensor network-based agricultural monitoring system: A clustering compressed sampling approach," *IEEE Trans. Ind. Inform.*, vol. 17, no. 6, pp. 4411–4420, Jun. 2021.

[15] Z. Yao, W. Cheng, W. Zhang, and H. Zhang, "Resource allocation for 5G-UAV based emergency wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3395–3410, Nov. 2021.

[16] Q. Guo *et al.*, "Minimizing the longest tour time among a fleet of UAVs for disaster area surveillance," *IEEE Trans. Mobile Comput.*, early access, Nov. 16, 2020, doi: 10.1109/TMC.2020.3038156.

[17] Z. Ning *et al.*, "5G-enabled UAV-to-community offloading: Joint trajectory design and task scheduling," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3306–3320, Nov. 2021.

[18] H. Peng and X. Shen, "Multi-agent reinforcement learning based resource management in MEC- and UAV-assisted vehicular networks," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 131–141, Jan. 2021.

[19] M. Samir, S. Sharafeddine, C. M. Assi, T. M. Nguyen, and A. Ghrayeb, "UAV trajectory planning for data collection from time-constrained IoT devices," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 34–46, Jan. 2020.

[20] M. Gapeyenko, D. Moltchanov, S. Andreev, and R. W. Heath, "Line-of-sight probability for mmWave-based UAV communications in 3D urban grid deployments," *IEEE Trans. Wireless Commun.*, vol. 20, no. 10, pp. 6566–6579, Oct. 2021.

[21] M. Shen, A. Liu, G. Huang, N. N. Xiong, and H. Lu, "ATTDC: An active and traceable trust data collection scheme for industrial security in smart cities," *IEEE Internet Things J.*, vol. 8, no. 8, pp. 6437–6453, Apr. 2021.

[22] Y. Zhou *et al.*, "Secure communications for UAV-enabled mobile edge computing systems," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 376–388, Jan. 2020.

[23] Q. Li and L. Yang, "Beamforming for cooperative secure transmission in cognitive two-way relay networks," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 130–143, 2020, doi: 10.1109/TIFS.2019.2918431.

[24] Z. Ning *et al.*, "Blockchain-enabled intelligent transportation systems: A distributed crowdsensing framework," *IEEE Trans. Mobile Comput.*, early access, May 13, 2020, doi: 10.1109/TMC.2021.3079984.

[25] C. Miguel *et al.*, "Practical Byzantine fault tolerance," in *Proc. 3rd Symp. Operating Syst. Des. Implementation*, New Orleans, USA, 1999, vol. 99, pp. 173–186.

[26] D. Kozhaya, J. Decouchant, V. Rahli, and P. Esteves-Verissimo, "PISTIS: An event-triggered real-time Byzantine-resilient protocol suite," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 9, pp. 2277–2290, Sep. 2021.

[27] S. Zhang, H. Zhang, B. Di, and L. Song, "Cellular UAV-to-X communications: Design and optimization for multi-UAV networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 2, pp. 1346–1359, Feb. 2019.

[28] M. Morelli, C. Kuo, and M. Pun, "Synchronization techniques for orthogonal frequency division multiple access (OFDMA): A tutorial review," *Proc. IEEE Proc. IRE*, vol. 95, no. 7, pp. 1394–1427, Jul. 2007.

[29] S. Doğan, A. Tusha, and H. Arslan, "NOMA with index modulation for uplink URLLC through grant-free access," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 6, pp. 1249–1257, Oct. 2019.

[30] C. Lorenzo, R. Christian, and G. Per, "Age of information-aware scheduling for timely and scalable Internet of Things applications," in *Proc. IEEE Conf. Comput. Commun.*, 2019, pp. 2476–2484.

[31] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled Internet of Vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2906–2920, Mar. 2019.

[32] T. Amila, N. Peter, Z. Banaszak, and B. Grzegorz, "Energy consumption in unmanned aerial vehicles: A review of energy consumption models and their relation to the UAV routing," in *Proc. Int. Conf. Inf. Syst. Architecture Technol.*, 2018, pp. 173–184.

[33] Z. Zhou *et al.*, "Energy-efficient industrial Internet of UAVs for power line inspection in smart grid," *IEEE Trans. Industrial Informatics*, vol. 14, no. 6, pp. 2705–2714, 2018.

[34] R. Marler and J. Arora, "Survey of multi-objective optimization methods for engineering," *Struct. Multidisciplinary Optim.*, vol. 26, no. 6, pp. 369–395, 2004.

[35] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*.

[36] Z. Ning *et al.*, "Online scheduling and route planning for shared buses in urban traffic networks," *IEEE Trans. Intell. Transp. Syst.*, early access, Mar. 24, 2021, doi: 10.1109/TITS.2020.3036396.

[37] E. Martins, "On a multicriteria shortest path problem," *Eur. J. Oper. Res.*, vol. 16, no. 2, pp. 236–245, 1984.

[38] Z. Ning *et al.*, "Intelligent edge computing in Internet of Vehicles: A joint computation offloading and caching solution," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 4, pp. 2212–2225, Apr. 2021.

[39] S. M. R. Islam, N. Avazov, O. A. Dobre, and K.-s. Kwak, "Power-domain non-orthogonal multiple access (NOMA) in 5G systems: Potentials and challenges," *IEEE Commun. Surv. Tut.*, vol. 19, no. 2, pp. 721–742, Apr.–Jun. 2017.

[40] X. Che, I. Barry, and L. Ling, "A survey of current YouTube video characteristics," *IEEE MultiMedia*, vol. 22, no. 2, pp. 56–63, Apr.–Jun. 2015.

[41] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Performance optimization for blockchain-enabled industrial Internet of Things (IIoT) systems: A deep reinforcement learning approach," *IEEE Trans. Ind. Inform.*, vol. 15, no. 6, pp. 3559–3570, Jun. 2019.

[42] S. Kung, S.-C. Lo, and Lewis, "Optimal systolic design for the transitive closure and the shortest path problems," *IEEE Trans. Comput.*, vol. C-36, no. 5, pp. 603–614, May 1987.

[43] Z. Ning *et al.*, "When deep reinforcement learning meets 5G-enabled vehicular networks: A distributed offloading framework for traffic Big Data," *IEEE Trans. Ind. Inform.*, vol. 16, no. 2, pp. 1352–1361, Feb. 2020.

Zhaolong Ning (Senior Member, IEEE) received the Ph.D. degree from Northeastern University, China in 2014. He was a Research Fellow with Kyushu University, Japan, from 2013 to 2014. He is currently a Full Professor with the College of Communication and Information Engineering, the Chongqing University of Posts and Telecommunications, Chongqing, China. He has authored or coauthored more than 120 scientific papers in international journals and conferences. His research interests include Internet of Things, mobile edge computing, deep learning, and resource management. Dr. Ning is an Associate Editor or the Guest Editor of several journals, such as IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON SOCIAL COMPUTATIONAL SYSTEMS, *The Computer Journal* and so on.

Handi Chen (Graduate Student Member, IEEE) received the B.Sc. degree in network engineering from the Tianjin University of Science and Technology, Tianjin, China, in 2019. She is currently working toward the M.Sc. degree with the School of Software, Dalian University of Technology, Dalian, China. Her research interests include mobile edge computing, resource allocation, and network optimization.

Xiaojie Wang (Member, IEEE) received the Ph.D. degree from the Dalian University of Technology, Dalian, China, in 2019. After that, she was a Postdoctor with the Hong Kong Polytechnic University. She is currently a Distinguished Professor with the College of Communication and Information Engineering, the Chongqing University of Posts and Telecommunications, Chongqing, China. She has authored or coauthored more than 40 scientific papers in international journals and conferences, such as IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, and IEEE COMMUNICATIONS SURVEYS AND TUTORIALS. Her research interests include wireless networks, mobile edge computing, and machine learning.

Shupeng Wang (Member, IEEE) received the M.S. and Ph.D. degrees from the Harbin Institute of Technology, Harbin, China, in 2004 and 2007, respectively. He is currently a Senior Engineer with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China. His research interests include big data management and analytics, network storage, etc.

Lei Guo received the Ph.D. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2006. He is currently a Full Professor with the Chongqing University of Posts and Telecommunications, Chongqing, China. He has authored or coauthored more than 200 technical papers in international journals and conferences. His current research interests include communication networks, optical communications, and wireless communications. He is the Editor for several international journals.