Verifiable Event Record Management for a Store-Carry-Forward-Based Data Delivery Platform by Blockchain

Yoshito Watanabe[1], Wei Liu[1], Alhabib Abbas[2], Yiannis Andreopoulos[2], Mikio Hasegawa[3], and Yozo Shoji[1]

[1]*Social-ICT System Laboratory*
National Institute of Information and Communications Technology
4-2-1, Nukui-Kitamachi, Koganei, Tokyo 184-8795, Japan
{yoshito-watanabe, wei_liu, shoji}@nict.go.jp

[2] *Dept. of Electronic and Electrical Engineering*
University College London
Torrington Place, London, WC1E 7JE, UK
{alhabib.abbas.13, i.andreopoulos}@ucl.ac.uk

[3] *Dept. of Electrical Engineering*
Tokyo University of Science
6-3-1 Niijyuku, Katsushika-ku, Japan
hasegawa@ee.kagu.tus.ac.jp

***Abstract*—We propose a novel database management framework for data delivery services based on Store-Carry-Forward (SCF) techniques. The platform we present consists of heterogeneous wireless opportunistic networks of long-range narrowband and short-range broadband communications. We introduce a blockchain-based method by which to verify the record of delivery events on a decentralized network. A new consensus mechanism named proof-of-forwarding (PoF) is proposed to substitute the function of previously proposed proof-of-work (PoW) methods, while significantly improving computational complexities of block generation. Specifically, in our proposal a block is generated exclusively when data delivery agents perform node-to-node direct communication using a short-range high-speed wireless standard to deliver data. We additionally propose a digital signature overlay to prevent malicious nodes from producing fake transactions without any effort to carry data content to recipients. Simulation results show that our blockchain-based framework robustly manages data delivery records, where 97% of Distributed Denial-of-Service (DDoS) attacks can be prevented even when half of the entire nodes are assumed to be malicious.**

***Index Terms*—Blockchain, store-carry-forward technique, digital signature, heterogeneous network, proof-of-forwarding**

I. Introduction

Nowadays, various kinds of communication standards are available in the market, of which 5G and Beyond-5G (B5G)/6G are attracting much attention owing to their capability of high-speed and ultra-reliable low-latency communication (URLLC). However, the maintenance of base stations demands both time and money, consequentially impeding the expansion of coverage areas to some countryside or depopulated areas.

On the other hand, store-carry-forward (SCF) techniques [1], [2], which facilitate data transfer through physical nodes over traversing opportunistic networks, have the potential to achieve the higher throughput than those of typical internet connections or cellular networks. We have demonstrated that this is especially true in contexts where extremely high-speed wireless standards, e.g., millimeter-wave (mmW) communication systems, are available for direct node-to-node communications, naming the networking concept as "piggy-back network" [3]. Motivated by this, we promote the construction of a data delivery platform utilizing already-existing mobile and fixed facilities, e.g., vehicles and beverage vending machines in a city in Japan [4].

978-1-7281-4490-0/20/$31.00 © 2020 IEEE

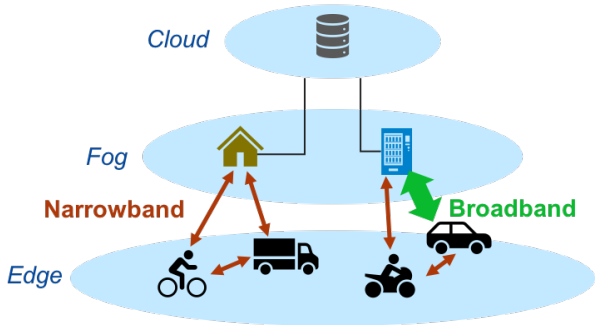

Fig. 1. A conceptualized system model of the proposed service platform using heterogeneous wireless standards.

A conceptualized system model is shown in Fig. 1, where the edge nodes are opportunistically connected by the heterogeneous network of long-range narrowband and short-range, but extremely high-speed, broadband communication standards through fog and cloud machines. Specifically, narrowband communications are utilized for controlling the nodes as well as transmitting small-volume data, and broadband communications are utilized for the transmission of large-volume data.

In the proposed platform, users who contribute to delivering data should obtain rewards. However, some malicious users may try to earn rewards illegally, and thus the service designers could be involved in a lot of concerns to prevent fraud. For example, if a service relies on centralized cloud servers to collect the transactions, we should continuously update the software and prepare measures against abrupt shutdown, intrusion, and falsification of the database. A blockchain technology, which is a protocol-based database management method, can be a solution to storing the transactions on a decentralized system securely and cost-effectively.

Proof-of-work (PoW) is a well-known consensus algorithm to generate a block in a conventional blockchain network [5]. The task of PoW is to find a number-used-once (nonce) by solving a hash puzzle. One of the problems in PoW is that an estimated nonce is not useful for anything except verifying a blockchain. Other consensus mechanisms have been proposed, where the computational resources are used beneficially for conducting valuable tasks. In Primecoin [6], finding prime numbers is the task of generating a block instead of conventional PoW. There have been other approaches where miners train machine learning models to generate a block [7], [8]. However, we consider that readily-available devices on the market like off-the-shelf smartphones, where the hardware specs are limited, are employed as the node devices in the proposed platform. Therefore, the above computation-depended consensus mechanisms cannot be straightforwardly applied to our system. There have been other consensus algorithms such as proof-of-stake (PoS) used in Ethereum [9] and proof-of-importance (PoI) used in NEM [10], but they could produce some bias in authority in the network. Accordingly, another consensus mechanism that is suitable for the proposed platform is required.

This paper presents a novel framework to manage verifiable event record for an SCF-based data delivery platform by blockchain. Assuming that a data provider generates large-volume data contents and several recipients ask for the data to be delivered, mobile agents deliver the data to each recipient by SCF techniques. A new consensus mechanism named proof-of-forwarding (PoF) is proposed such that the blockchain becomes more suitable for our SCF-based data delivery platform. We additionally propose a digital signature overlay to prevent malicious nodes from producing fake transactions and obtaining rewards fraudulently. Computer simulations were conducted to demonstrate the robustness of delivery records against the attacks where malicious nodes try to falsify the records of honest nodes.

II. System Model

This section describes the system model we propose upon the concept illustrated in Fig. 1.

A. An Overview of a Data Delivery System Based on SCF Principle

Figure 2 shows the proposed data dissemination system which is based on SCF techniques. There are three types of nodes named *provider*, *deliverer*, and *recipient*. A provider generates large data contents, which should be delivered to the recipients. Deliverers, often referred to as *data carriers* or *data mules* in the literature, are mobile nodes to conduct SCF techniques; they first download the copy of data content from the provider when they approach, then carry it to the location of the recipients, and upload it to the recipients. All the nodes have their unique account address generated by, e.g., SHA-256, which is a commonly employed hash function in this paper for the purpose of generating hash values.

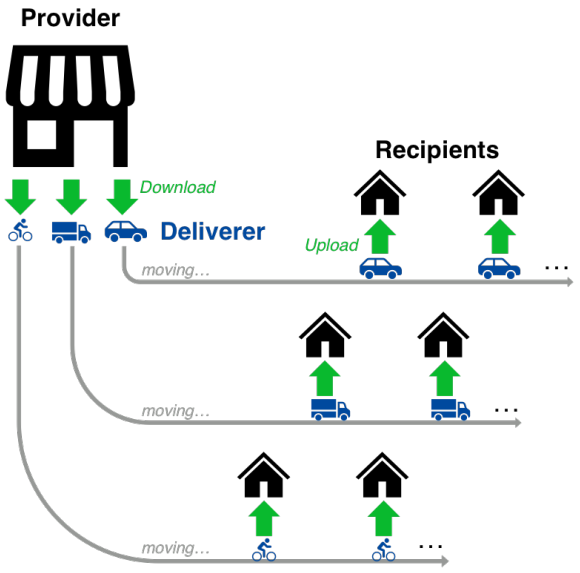

Fig. 2. An SCF-based data dissemination system.

The size of data content is assumed to be tens or hundreds of gigabytes or even terabytes. There are several recipients for one data content. Once data is generated, its digest information and the hash key are assumed to be disseminated in realtime throughout the network using the narrowband communications. In this way, the recipients can know their desired contents beforehand.

The generation of contents is a stochastic process in practice, and the number of providers is assumed to be comparatively small throughout the network.

B. Managing Event Record by Blockchain

We employ a blockchain technology to manage verifiable delivery record in the network and to prevent malicious nodes from overwriting the record. We propose a new consensus mechanism, named *proof-of-forwarding (PoF)*, which considers the transmission of data contents through direct communications with a short-range wireless standard between nodes as the consensus in a blockchain. All the transitions of data contents are represented as *transactions*. There are three types of transactions: GEN, Tx and Rx transactions. Transactions are propagated throughout the platform by the aforementioned narrowband network once they are produced. All nodes hold all transactions that are issued in the network. These transactions will be recorded in the blockchain. All nodes in the platform are assumed to have a full blockchain for convenience.

We also propose a digital signature overlay that can validate the direct connections between deliverers and recipients. The details of the proposed methods are described in Section III and IV.

Note that the incentive design, that is, how much incentive will be paid and who will pay it, should be considered depending on the services, which is beyond the scope of this paper.

TABLE I
BASIC ITEMS IN A TRANSACTION.

Field	Description
Trans. type	The type of the transaction, i.e., GEN, Tx, or Rx.
Content hash	The hash value of data content. By this, the content associated with the transaction can be identified.
Sender Account	The account address of the data sender.
Receiver Account	The account address of the data receiver.
Signature	Digital signature produced by the receiver using digital signature overlay.
Timestamp	Unix timestamp when the transaction is generated.

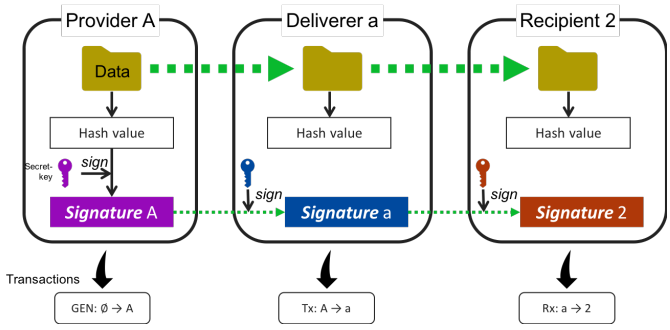

Fig. 3. A flow of digital signature overlay. The transactions to be generated by each node are also shown, in which the form of X: Y → Z represents the transaction type X of data transition from sender Y to receiver Z.

III. TRANSACTIONS AND DIGITAL SIGNATURE OVERLAY

This section explains one of the proposed techniques, i.e., digital signature overlay, which is employed to ensure both the identification of a deliverer and the fact that he/she completes delivering the data to the recipient.

A. Transactions

A transaction is like a digital container to store the items that are required to validate the delivery record in a specific format. The proposed SCF-based platform operates via the following three processes:

- A provider *generates* data (GEN)
- A provider *transmits* data to a deliverer (Tx)
- A recipient *receives* data from a deliverer (Rx)

The terms in the parentheses above are the types of processes, namely, the types of transactions.

Table I lists the basic items required to trace whole records in the network. Every transaction should be associated with specific data content, where the content hash identifies which content the transaction is associated with. Signatures are generated by data receivers using digital signature overlays, which we further detail in Section III-B. Note that, for GEN transactions, there is no data transmission process, and thus sender accounts in every GEN transaction are set to a predefined constant (e.g., "0").

B. Digital Signature Overlay

Digital signatures [11] have been used conventionally to ensure the creator of data by encoding the digest value of data with the public-key cryptography. We employ this technology to verify the deliverer's delivering record to the recipient.

Figure 3 illustrates the flow of data transition from provider A to recipient 2 through deliverer a using the proposed *digital signature overlay*. The transactions to be generated by each node are also shown in the figure, in which the form of X: Y → Z represents the transaction type X of data transition from sender Y to receiver Z. As a premise, all the individual nodes in the network have their own pairs of a public-key and secret-key generated by the elliptic curve digital signature algorithm (ECDSA) [12].

The first step of the proposed method is performed in a conventional way; the provider encodes the hash value of the data content using the provider's secret-key. The provider then transmits the signature itself as well as the data content when the deliverer approaches the provider.

After the deliverer receives the content and the signature from the provider, it further signs the signature (Signature A in Fig. 3) by the deliverer's secret-key and generates an overlaid signature (Signature a).

The same process is carried out on the recipient side; after the recipient receives the signature as well as the data content from the deliverer, it signs the already-overlaid signature using the recipient's secret-key and generates a doubly-overlaid signature (Signature 2 in Fig. 3). Every signature produced by this method is included in a transaction.

Anyone in the network can perform the verification of the overlaid signatures and its process is as follows: We first decode the doubly-overlaid signature (Signature 2 in Fig. 3) by the recipient's public-key, where the resulting value should be completely equal to the single-overlaid signature (Signature a in Fig. 3). Then, we decode the single-overlaid signature by the deliverer's public-key, and the resulting value should be equal to the original signature (Signature A in Fig. 3).

Consequently, the recipient's own approval using his/her secret-key is indispensable to issue an Rx transaction. Therefore, the digital signature overlay can prevent malicious nodes from producing fake transactions without any effort to carry data to the recipients.

IV. PROOF-OF-FORWARDING: A NOVEL CONSENSUS MECHANISM FOR SCF-BASED DATA DELIVERY SERVICES

Generally in blockchain technologies, the consensus mechanisms should be hard to be conducted but easy to be verified. Conventional PoW in Bitcoin's blockchain has this characteristic; resolving hash puzzle to find a nonce is a much hard task but the verification process is easily performed by applying a hash function to the block's components. We propose a new consensus mechanism, PoF, that meets with the above requirement and is compatible with our opportunistic data delivery platform.

This paper assumes that only deliverers can generate a block. The proposed PoF operates as follows: When a deliverer finishes delivering, or *forwarding*, data to a recipient by node-to-node direct communication, the recipient issues an Rx transaction including the overlaid signature described in

Section III. This Rx transaction is sent back to the deliverer as the approval of generating a block. The deliverer then generates a block, adding all the transactions remaining in the network to the block.

Every block in our blockchain has one Rx transaction and other types of transactions[1]. As is the case with conventional blockchains, the hash value of a block is generated using a hash function. By inserting the previous block's hash value to the block header, we can chain the blocks. We can verify the blockchain by comparing the previous block's hash value contained in a block header with the hash value obtained by actually applying a hash function to the previous block. The longer a chain becomes, the more robust the database is. This property enables us to ensure that no duplicate data content can be delivered to the same recipient if the original Rx transaction is already included in the blockchain.

As mentioned above, generating a block in PoF is not easy because the deliverers carry out the SCF-based data delivery by physically moving and approaching a recipient, of which evidence is confirmed by the digital signature overlay. On the other hand, the verification process of PoF is easy to be conducted. Unlike the task to find a nonce in conventional PoW, the task to deliver data itself contributes to activating the services on the platform. Furthermore, moving speeds of deliverers and communication power should be limited by road traffic laws and radio laws, respectively; whereas it is possible for miners in conventional blockchains to increase the computational power used for PoW inexhaustibly if they can spend the money. Eventually, the proposed framework could more discourage malicious nodes from accomplishing Distributed Denial-of-Service (DDoS) attacks than PoW-based blockchain systems.

V. Computer Simulations

We evaluated the robustness of the proposed framework against DDoS attacks from malicious nodes by computer simulations. Since the application of the proposed framework is limited to our specific scenario, the comparison with conventional blockchain-based schemes is beyond the scope of this paper.

A. Attack Scenario

We assume that malicious deliverers will overwrite the delivery records that have been originally accomplished by honest deliverers. To achieve this attack, malicious nodes try to grow another branch separately from the honest branch as the blockchain considers the longer branch to be valid. In our simulations, these two branches share only the genesis block[2], constituting different branches from the second blocks.

We assume that malicious nodes have their own network that is isolated from the network managed by honest nodes. Furthermore, it is assumed that they are able to issue a new Rx transaction by forcing duplicate content to recipients who already have the same content delivered by honest nodes such that the delivery record is overwritten. These forcefully-produced Rx transactions are shared only among the network of malicious nodes. Malicious nodes are also able to issue Rx transactions *legally*, where a malicious node is the first one to deliver data to a recipient. These legally-produced Rx transactions as well as all GEN and Tx transactions of malicious nodes are broadcasted to both the malicious and honest network. Note that, though this situation should be impossible practically, it would be possible if both deliverers and recipients are malicious.

On the other hand, the transactions of honest nodes exist only in their own network. It is impossible for honest nodes to overwrite the data contents of the recipients where malicious nodes originally delivered the same contents. Therefore, a portion of Rx transactions of malicious nodes subsists in the honest branch even if the attack fails.

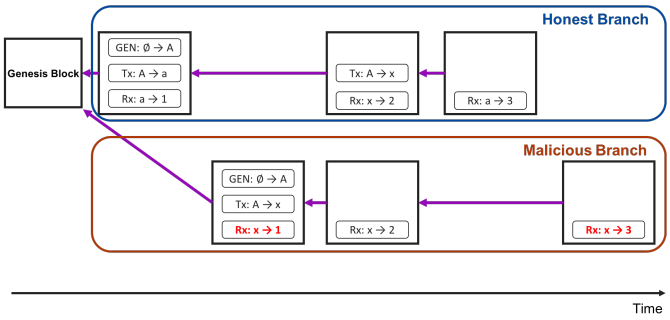

Fig. 4. An example of branched blockchain, where a malicious node is denoted by x. The delivery records to recipient 1 and 3 are overwritten in the malicious branch, but that to recipient 2 by the malicious node cannot be overwritten in the honest branch.

Figure 4 illustrates an example of a branched blockchain, where only the transactions are depicted as block's components for simplicity. A malicious deliverer is denoted by x in the figure. From the second block (counting from the genesis block) in the honest branch, we can confirm that deliverer a finished delivering data to recipient 1. However, deliverer x has overwritten the delivery record to recipient 1 in the malicious branch. The legally-generated Rx transaction from deliverer x to recipient 2 can be stored in both branches, which cannot be overwritten by honest nodes. One can notice that the length of the malicious branch never exceeds that of the honest branch. Therefore, we determine that the attack by malicious nodes is successful if the number of blocks in the malicious branch is equal to that in the honest branch.

B. Simulation Conditions

We built a mobility model on a two-dimensional square plane based on a modified random waypoint model for the deliverers' mobility: First, all deliverers are positioned at random locations, and then they head for a provider. After they reach the provider and stop for a certain period to download data, they move to the recipients in order from the nearest one who has not yet received the data. When another deliverer

[1]Practically, we can employ Merkle trees to store the transactions in a block efficiently.

[2]The root block of blockchain where no meaningful information is included.

TABLE II
SIMULATION PARAMETERS.

Area of simulation	2000×2000 m^2
Total number of deliverers	100
Number of providers	10
Number of recipients for one content	10
Distribution of velocity of honest deliverers v_h	$U(1, 15)$ m/s
Stop time when a deliverer reaches each destination	10 seconds

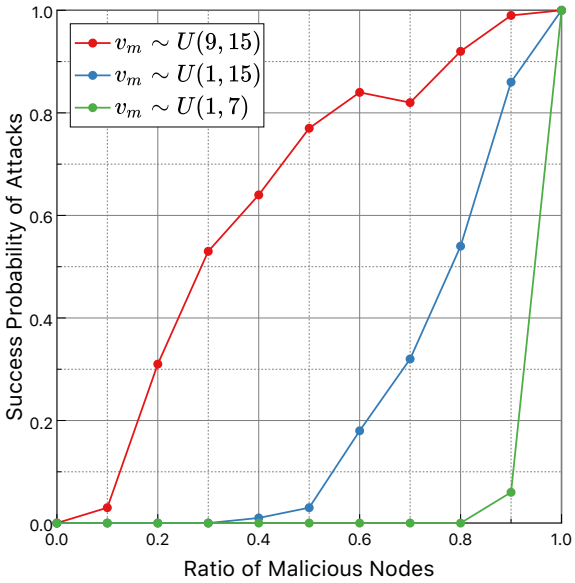

Fig. 5. Success probability of attacks vs. ratio of malicious nodes among deliverers, changing the distributions of moving velocities of malicious nodes to $U(9, 15)$, $U(1, 15)$, and $U(1, 7)$ m/s.

finishes the delivery to the current-target recipient during the move, it changes the direction to the other nearest recipient. The velocities of the deliverers are given by random variables and vary after every stop. One provider generates one content, and its location is random. For simplicity of analysis, we assume all deliverers cannot download a different content from a different provider until one content is disseminated to all the recipients.

Table II summarizes the simulation parameters. We denote the velocity of honest deliverers by v_h, and it conforms to a uniform distribution, i.e., $v_h \sim U(1, 15)$. Note that the stop time when a deliverer reaches each destination is set to 10 seconds; this is equivalent to the delivery of data of 50 gigabits if we assume the throughput of 5 Gbps for node-to-node broadband communications. Based on our simulation parameters, totally 101 blocks including a genesis block should be generated.

C. Results

We evaluated the success probability of malicious nodes' DDoS attacks by changing the ratio of malicious nodes among all deliverers. Specifically, we compared the growing speed of honest and malicious branches, and if the malicious branch is faster, the attack is considered to be successful. The success probability of the attack is computed by iterating the simulation of the same conditions over 100 times.

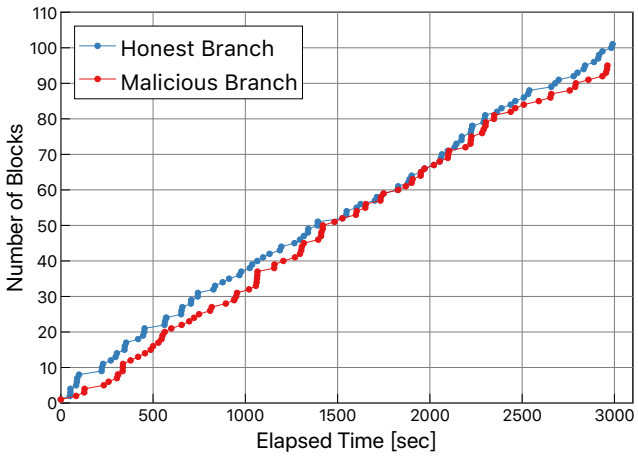

Fig. 6. The lengths of honest and malicious branches when $v_m \sim U(1, 15)$ and the ratio of malicious nodes is 50%.

Figure 5 shows the relationship between the ratio of malicious nodes and the success probability of DDoS attacks, changing the distributions of moving velocities of malicious nodes, denoted by v_m, to $U(9, 15)$, $U(1, 15)$, and $U(1, 7)$ m/s.

In the case of $v_m \sim U(1, 15)$, which is equivalent to the distribution of v_h, the success probability of attacks is only 3% when the ratio of malicious nodes equals 50%. This is because the Rx transactions legally generated by malicious nodes, regardless of the effort made by malicious nodes, contribute to growing the honest branch, which is inevitable in our scenario. Therefore, it is hard for malicious nodes to achieve the DDoS attack only by increasing malicious nodes. In contrast, in conventional PoW-based blockchains, attacks should be mostly successful if the computational power of malicious nodes exceeds the majority of the entire network (i.e., 51% attack). To achieve more than 50% of success probability of attacks on the proposed platform, approximately 80% of malicious nodes are needed when all nodes have the same velocity distribution.

Figure 6 shows a comparison of the lengths between honest and malicious branches for one trial of simulations when $v_m \sim U(1, 15)$ and the ratio of malicious nodes is 50%. Both performances are relatively fair, but the honest branch always outperforms the malicious branch in terms of the chain length due to the aforementioned property where the legally-generated Rx transactions by malicious nodes contribute to growing the honest branch.

The velocity of malicious nodes, v_m, is also a factor in the success of attacks in addition to the number of nodes. It is obvious from Fig. 5 that the success probability of attacks increases as the moving velocity increases; when $v_m \sim U(1, 7)$ m/s, the attack is hardly achieved until the ratio of malicious nodes equals 90%. However, when $v_m \sim U(9, 15)$, 77% of attacks are successful at the ratio of malicious nodes equal to 90%. The success probability of attacks increases to

99% when the ratio of malicious nodes is equal to 90%. Nevertheless, as discussed in Section IV, the velocity is generally regulated by road traffic laws in public, and thus such a situation would be unlikely to happen in practice.

VI. Application Scenarios

Several application scenarios on the proposed platform illustrated in Fig. 1 are considered. One possible application is a delivery service of large-volume data. While video streaming services, like YouTube[3] and Netflix[4] are now much popular worldwide, there is a recent survey result [13] that the half number of home internet in Japan is in fact for mobile phone lines, of which throughput could be poor due to the monthly limitations of data transmission. We can also see that, from the same survey result, internet users among people over 70-years-old is less than 50%. For such users, we can deliver the data of rich-contents, such as 4K and 8K movies or video messages from local social welfare councils.

Another possible application is an automated vision-sensing service for city surveillance, where social events are predicted by mobile and fixed cameras exploiting artificial intelligence (AI) technologies. Our work can complement previously proposed action recognition bitstreams [14], where only necessary information extracted on resource-limited edge devices is transmitted to resource-rich fog devices through narrowband networks in advance to predict events. Then, the broadband communication is utilized for transmitting the entire video frames in an SCF manner if desired. In this way, we can build the big video data collection of the socially valuable events in a community, which could be provided for tourists or utilized for the deterrence against city crimes.

VII. Conclusions

This paper proposed a framework to store the event record on an SCF-based data delivery platform in a verifiable manner by blockchain. A new consensus mechanism, named PoF, produces a block based on a node-to-node communication log and is executable using IoT devices with limited computing power. A digital signature overlay method was also proposed to ensure the evidence of deliverers' physical movement to the locations of data recipients and their data forwarding process. The simulation results showed that the database was quite robust against DDoS attacks; only 3% of attacks were successful even when half of the entire nodes were malicious and they moved at the same speed as honest nodes. The proposed framework could highly discourage malicious nodes from executing attacks since malicious nodes need to be able to move quickly or physically increase the number of their entities. We also provided several examples of future application scenarios.

This study did not consider the propagation delay of the transactions and blocks over opportunistic networks. More elaborate analysis considering this delay is necessary, which is left as our future work. Besides, the channel design for heterogeneous wireless systems using both narrowband and broadband communications will be necessary.

[3]https://www.youtube.com

[4]https://www.netflix.com

References

[1] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic networking : Data forwarding in disconnected mobile ad hoc networks," *IEEE Communications Magazine*, vol. 44, no. November, pp. 134–141, 2006.

[2] S. Trifunovic, S. T. Kouyoumdjieva, B. Distl, L. Pajevic, G. Karlsson, and B. Plattner, "A Decade of research in opportunistic networks : Challenges , relevance , and future directions," *IEEE Communications Magazine*, vol. 55, no. January, pp. 168–173, 2017.

[3] Y. Shoji, W. Liu, and Y. Watanabe, "Community-based "Piggy-back Network" utilizing local fixed & mobile resources supported by heterogeneous wireless & AI-based mobility prediction," in *IEEE Vehicular Technology Conference (VTC) Workshops*, May 2020.

[4] Y. Watanabe, W. Liu, and Y. Shoji, "A Demonstrative Study on the Potential of Store-Carry-Forward-Based Contents Delivery by a Beverage Supplier's Logistics Network," in *Proceedings of 2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct. 2019, pp. 65–70.

[5] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," p. 9, 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[6] S. King, "Primecoin: Cryptocurrency with Prime Number Proof-of-Work," p. 6, Jul. 2013. [Online]. Available: https://primecoin.io/bin/primecoin-paper.pdf

[7] C. Chenli, B. Li, Y. Shi, and T. Jung, "Energy-recycling Blockchain with Proof-of-Deep-Learning," in *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, May 2019, pp. 19–23.

[8] F. Bravo-Marquez, S. Reeves, and M. Ugarte, "Proof-of-Learning: A Blockchain Consensus Mechanism Based on Machine Learning Competitions," in *2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON)*, Apr. 2019, pp. 119–124.

[9] W. Gavin, "Ethereum: A Secure Decentralised Generalised Transaction Ledger," 2018. [Online]. Available: https://ethereum.github.io/yellowpaper/paper.pdf

[10] "NEM Technical Reference," Feb. 2018. [Online]. Available: https://nem.io/wp-content/themes/nem/files/NEM_techRef.pdf

[11] R. C. Merkle, "A Certified Digital Signature," in *Advances in Cryptology — CRYPTO' 89 Proceedings*, ser. Lecture Notes in Computer Science, G. Brassard, Ed. New York, NY: Springer, 1990, pp. 218–238.

[12] D. Johnson, A. Menezes, and S. Vanstone, "The Elliptic Curve Digital Signature Algorithm (ECDSA)," *International Journal of Information Security*, vol. 1, no. 1, pp. 36–63, Aug. 2001.

[13] Ministry of Internal Affairs and Communications, ""Communications Usage Trend Survey" in 2017 Compiled," May 2018. [Online]. Available: https://www.soumu.go.jp/johotsusintokei/tsusin_riyou/data/eng_tsusin_riyou02_2017.pdf

[14] M. Jubran, A. Abbas, A. Chadha, and Y. Andreopoulos, "Rate-Accuracy Trade-Off in Video Classification With Deep Convolutional Neural Networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 1, pp. 145–154, Jan. 2020.