This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3118928, IEEE Internet of Things Journal

Fast and Scalable Distributed Consensus over Wireless Large-Scale Internet-of-Things Network

Hojung Lee, *Graduate Student Member, IEEE,* Hyowoon Seo, *Member, IEEE,* Wan Choi, *Fellow, IEEE,*

***Abstract*—Due to the rapid paradigm shift in IoT networks from wired and centralized to flexible wireless and decentralized networks, building effective and reliable distributed consensus mechanisms over wireless is becoming essential. Especially, since the performance of consensus over communication endpoints in a large-scale wireless network is limited by their communication capability, it requires a careful co-design of communication and consensus to attain a fast and scalable distributed wireless consensus mechanism with high resiliency against faulty nodes. Within this context, this paper addresses such problem by designing two wireless consensus mechanisms that well-suit in large-scale wireless networks. On the one hand, as a reinterpretation of the conventional referendum consensus (RC) in a large-scale wireless network, *gossip-broadcasting based referendum consensus (GB-RC)* is proposed. On the other hand, to overcome the scalability issue of the GB-RC, *cooperative-broadcast based electoral-college consensus (CB-EC)* is proposed. By mathematically analyzing the performance of both of the consensus mechanisms, in terms of consensus latency and resiliency against the faulty nodes, we show that the GB-RC outperforms the conventional RC, while the CB-EC significantly reduces the consensus latency compromising the stochastic resiliency. We further evaluate their performance numerically to show their effectiveness and feasibility under realistic large-scale wireless environments.**

***Index Terms*—Large-scale wireless network, gossip-broadcasting, cooperative-broadcasting, wireless distributed consensus, Byzantine fault tolerance (BFT).**

I. Introduction

The rapid development of wireless communication technology has enabled us to live in the Internet of Things (IoT) world, where almost everything is connected. At the core of IoT, connecting multiple devices enables integrated control to accomplish tasks that were impossible to be done by a device alone. Thus, in the past, it was a major concern in academia and industry to improve connectivity between the devices [1]–[4], what until now is still of importance. However, as the connectivity of IoT has improved, the number of devices entering the IoT network has increased tremendously, thereby making it difficult to control them. Moreover, the devices are becoming surprisingly intelligent, leading a paradigm shift of IoT from the traditional central and rigid to novel distributed and flexible architectures, which makes number of ongoing tasks in one IoT network also increase in proportion to the increase of devices.

Hence, the cutting edge IoT networks call for well-designed autonomous and distributed consensus mechanisms among the devices therein, in order to provide cross-task coordination as well as intra-task coordination, while keeping out faulty decisions at individual IoT devices when accomplishing tasks. For example, in multitask-oriented IoT applications, such as autonomous cars, swarm drones, smart homes, factories, agriculture, stadium and communities [5]–[9], distributed consensus on the sequence of devices' actions brings better system stability and reliability. The problem is that, the existing distributed consensus mechanisms are ill-suited for wireless IoT networks, since the physical limitations of wireless communications, such as channel fading, noise, interference and limited transmission power, are not carefully considered. Specifically, such hardship will be especially amplified if the target application requires real-time operation, while the size of the network and the number and density of devices therein are large. Therefore optimized large scale consensus mechanism for wireless network will be necessary.

To address this problem, the main goal of this paper is to design a distributed consensus mechanism that makes fast and scalable consensus in the large-scale wireless IoT networks. Especially, we deal with a distributed consensus that tolerates faulty IoT devices. Accordingly, we propose two different approaches of distributed consensus mechanisms. The first one is a large-scale wireless referendum consensus that inherits the concept of the existing referendum consensus (RC) [10], [11] and utilizes the *gossip-broadcasting (GB)* message dissemination method, which is a mixture of two message dissemination methods, i.e., multi-hop one-to-one (gossiping) and single-hop one-to-many (broadcasting) methods. The proposed approach is termed *gossip-broadcasting based referendum consensus (GB-RC)*. The existing RC in [10] is a wireless implementation of PBFT [11] in two types, one with gossiping and the other with broadcasting. In [10], it is concluded that the one with broadcasting method is more efficient in small-scale network environment. However, the broadcasting method alone cannot guarantee the reliability of communication in the

Manuscript received Mar 24, 2021; revised Jun 30, 2021 and Sep 6, 2021; accepted Oct 2, 2021.

This research was supported in part by the Ministry of Science and ICT (MSIT), Korea, under the Information Technology Research Center (ITRC) support program (IITP-2020-0-01787) supervised by the Institute of Information & Communications Technology Planning & Evaluation (IITP) and in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2021R1A2C2003230).

H. Lee is with School of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, Korea (e-mail: hojung_lee@kaist.ac.kr).

H. Seo is with Centre for Wireless Communications, University of Oulu, Oulu 90014, Finland (e-mail: hyowoon.seo@oulu.fi).

W. Choi is with the Institute of New Media and Communications and Department of Electrical and Computer Engineering, Seoul National University (SNU), Seoul 08826, Korea (e-mail: wanchoi@snu.ac.kr). *(Corresponding author: Wan Choi)*

large-scale wireless networks, since nodes therein cannot reach all the other nodes with single-hop communication. Hence, the gossip-broadcasting method is utilized to enable the RC in large-scale wireless networks.

The second approach is a large-scale wireless representative consensus, termed *cooperative-broadcasting based electoral-college consensus (CB-EC)*, that leverages cooperative-broadcasting (CB) as a message dissemination method. The CB-EC is inspired by the U.S. presidential election process, in which the winner takes all the votes assigned for each state. Similarly, in the CB-EC, we locally conduct validation and the majority opinion is chosen as the representative opinion of the local network. To do this, the CB dissemination method plays an important role to enable the consensus process without any help from relays. In general, representative consensus methods, including the CB-EC, compromise the resiliency of the consensus process, so we further discuss the resiliency of the CB-EC.

The performance of the two proposed approaches are examined by their consensus latency. Compared to the existing broadcasting based RC, the proposed GB-RC outperforms them with lower latency in the large-scale wireless network. In particular, although the existing broadcasting based RC had good communication-efficiency in a small-scale network, there remained a scalability issue due to the physical limitation of message dissemination in a large scale network, and this problem was solved through GB dissemination method in the GB-RC. In addition, the CB-EC showed that the latency can be significantly reduced by compromising the resiliency of the consensus mechanism. Thus, the resiliency of the CB-EC is further investigated by measuring stochastic resiliency measure α*-resiliency*. Through this, it is shown that CB-EC is stochastically resilient to the faulty nodes under certain condition on the number of faulty nodes in the network, similar to the GB-RC.

A. Related work

The problem of data dissemination over wireless network has been widely studied in literature. For instance, data dissemination schemes concerning reliability of the dissemination were studied in [12], [13], using constructive interference between sensors in wireless sensor networks. Moreover, enhancing data throughput using hierarchical cooperation structure has been investigated in [14], and [15] has proposed a communication system termed *Choir* to improve the communication throughput and range in LP-WAN platform. However, the above mentioned studies were investigating only the data dissemination as communication technique, but did not consider higher level protocols, such as consensus mechanisms for wireless networks.

Meanwhile, the studies on distributed consensus mechanisms were also done for a long period time. Fundamental studies on the fault tolerance and reliability of distributed consensus mechanisms were mainly studied under the name of *Byzantine-Fault-Tolerance (BFT)* algorithms [11], [16]–[18], and *Paxos* algorithms [19], [20]. Moreover, resiliency against faulty nodes in asynchronous networks was analyzed [21]. Since these studies were considering a network architectures with a small number of nodes, further developments on the distributed consensus mechanisms for large-scale systems with low consensus latency were required, as the nowadays network size increased as well as the number of nodes therein.

In terms of the large-scale distributed consensus mechanism, *Blockchain* [22] is one of the most popular and widely used distributed ledger technology. However, since the Blockchain has its physical limitations, such as high consensus delays (e.g., several minutes–hours), the extent to which the mechanism can be applied is also limited. For example, some IoT applications which include mission-critical and real-time control aspects cannot leverage the Blockchain. Thus, from the consensus latency perspective, mechanisms based on permissioned network are gradually emerging as an appropriate solution for such applications. These mechanisms are designed based on the concepts of BFT algorithm [11], [18] that require exchanging voting information prior to the consensus process. Moreover, to overcome the scalability issues, sharding-based consensus mechanisms [23]–[27] were also studied, and recently, Blockchain technologies well-suited for IoT networks were investigated in [28], [29] and performance of directed acyclic graph (DAG)-based consensus algorithm under various network load conditions were analyzed in [30].

Nevertheless, most of the aforementioned studies on distributed consensus mechanisms assume that nodes communicate over fast and reliable wired links. To investigate theme over large-scale wireless networks, fundamental properties of wireless networks should be carefully considered and co-designed with consensus process. In this flow, several wireless distributed consensus mechanisms have recently been investigated in [10], [27], [31]–[35]. For instance, [31] proposed a wireless spectrum access scheme utilizing a Hashgraph-motivated consensus mechanism and a Byzantine consensus mechanism for permissionless wireless network was proposed in [27]. Yet, most of the preceding works on wireless distributed consensus mechanisms are application-specific, which do not reflect the fundamental relationship between wireless communication and consensus performance. In [10], a couple of consensus and communication co-designed mechanisms, were proposed, however they are only feasible in small-scale wireless networks. Therefore, in this paper, we expand the distributed consensus mechanisms proposed in [10] to be suitable to large-scale wireless networks, where direct communication between most distant nodes is not practically possible and the number of nodes therein is large, and further propose a novel communication-efficient distributed consensus mechanism via co-designing consensus and communication.

B. Contribution and Organization

The contributions of this paper are summarized as follows.

- We propose the *gossip-broadcasting based referendum consensus (GB-RC)* (Sec. III) by extending the conventional RC with broadcasting. The GB-RC leverages a communication method termed gossip-broadcasting (GB) (Sec. III-A), a combination of the gossiping and broadcasting, for effective and efficient message dissemination in a large-scale network.

- We propose a scalable and low latency consensus mechanism termed *cooperative-broadcasting based electoral-college consensus (CB-EC)* (Sec. IV), which is a communication and consensus co-designed consensus mechanism. The CB-EC leverages the cooperative-broadcasting method (Sec. IV-A) to have reduced consensus latency compared to the GB-RC.
- We derive the upper-bound of consensus latency of both GB-RC and CB-EC (Theorem 1 in Sec. III and Theorem 2 in Sec. IV). Furthermore, we analyze the stochastic resiliency of the CB-EC against faulty nodes, in terms of α-resiliency (Sec.IV-C). We further compare the performance of both GB-RC and CB-EC through mathematical analysis and numerical evaluation. For practical evaluation, we test the proposed mechanisms' effectiveness and feasibility in the real world wireless communication environment (Sec. V).

The remainder of this paper is organized as follows. In Sec. II, we explain the system architecture under study, including network model and channel model, in detail. In Sec. III, we propose and analyze the GB-RC and process of GB message dissemination method, and in Sec. IV we propose and analyze the CB-EC and CB message dissemination method. In Sec. V, we numerically evaluate the performance of the GB-RC and CB-EC by consensus latency and resiliency metric, followed by the conclusion in Sec. VI.

II. System Architecture

A. Network model

Consider a wireless square-shaped network with N fixed nodes, equipped with wireless transceivers and individual distributed ledger[1]. The nodes are located on the square grid of the network and suppose the distance between any two neighboring nodes as R. For the sake of convenience, define the set of nodes as $\mathcal{N} = \{1, \dots, N\}$, where the nodes are labeled in the order of their location in the network; from up-left to bottom-right. The network under study is permissioned, i.e., each node is assigned with a unique identifier and the other nodes know it. Though the nodes are independent and identical entities, however, depending on the roles of nodes during the consensus mechanism, the nodes are distinguished with different names, i.e., a proposer, validator and relay, hereafter. To illustrate, the *proposer* is a node that proposes new actions to the network, the *validator* is a node that validates the proposed actions and commits the validation result with the other validators to determine whether to accept the actions or not. Last, the *relay* is a node that relays the communication between two nodes that are located out of communication range of each other, thereby enabling multi-hop communication. Meanwhile, to overcome physical issues of wireless communication in a large-scale network, we consider that the network is divided into K small-scale square networks referred to as *local networks*. To avoid confusion, hereafter we call the original network before division as a *global network*.[2] The set of local networks is denoted by $\mathcal{K} = \{1, \dots, K\}$, where it is assumed that the local networks are labeled in the order of their location similar to the node label and $\sqrt{K}$ is a positive integer. Consequently, in each local network, there are D nodes in the network, where $\lceil N/K \rceil - 1 \leq D \leq \lceil N/K \rceil$ and the set of nodes in local network k is denoted by $\mathcal{N}_k$ for $k \in \mathcal{K}$. To avoid inter-local interference, we apply T-TDMA[3] method [37], where one local network communications are carried out, while $T-1$ surrounding local networks remain silent, and proceed by taking turns. Though TDMA itself does not give the best performance in terms of latency, however, we applied it to reflect the impact of interference mitigation via orthogonalization on latency performance in our analyses.

[1] A distributed ledger contains the series of valid actions and time stamps which enables to validate the proposed action as a state machine [36]. We assume the distributed ledgers are updated by a consensus mechanism and thus designing the mechanism is the main concern throughout this article.

B. Channel model

During the consensus process, the nodes disseminate information over wireless links, of which reliability is affected by large and small scale fading effects. Consider the channels experience path-loss with path-loss exponent η, which takes value between 2.7 and 3.5 for urban outdoor scenarios and from 1.6 to 3.3 for indoor scenarios, respectively. In detail, the path-loss between node i and j, for $i, j \in \mathcal{N}$ and $i \neq j$, is modeled as

$$\mathrm{PL}_{dB}(R_{ij}) = \mathrm{PL}_{dB}(R_0) + 10\eta \log_{10}\left(\frac{R_{ij}}{R_0}\right) \tag{1}$$

where $\mathrm{PL}_{dB}(R_0)$ is defined as the path loss at the reference distance R_0 and R_{ij} is the distance between the nodes i and j. The Rayleigh fading channel model is adopted since narrow-band communication and typical uniform scattering environments are under consideration. Let H_{ij} be the channel gain from node i to node j, following the complex normal distribution with zero mean and variance P_t (i.e., $H_{ij} \sim \mathcal{CN}(0, P_t)$). Moreover, we assume that the channel gains are independently and identically distributed (i.i.d.). Therefore the received signal-to-interference-plus-noise ratio (SINR) over the channel H_{ij} is obtained as

$$\mathrm{SINR}_{ij} = 10^{-\frac{\mathrm{PL}_{dB}(R_0)}{10}} \frac{|H_{ij}|^2 P_t}{P_n + I_{ij}} \left(\frac{R_0}{R_{ij}}\right)^{\eta} \tag{2}$$

where P_n is the additive white Gaussian noise (AWGN) power. I_{ij} is the inter-local interference power between node i and j under T-TDMA scheme and can be derived as

$$I_{ij} = \sum_{l=1}^{L} P_t |H_{I_l^i, j}|^2 \left(\frac{R_{I_l^i, j}}{R_0}\right)^{-\eta} \tag{3}$$

where $\mathcal{I}^i = \{I_1^i, I_2^i, ..., I_L^i\}$ is the set of L arbitrary nodes in different local networks transmitting messages in the same time slot with node i. Note that since we consider a directional message dissemination model, no interfering node is closer

[2] The shape of the global and local networks are assumed to be square for analytic simplicity and convenience, however, the proposed schemes in this article will still work with networks with other shapes such as circle and hexagon without loss of generality.

[3] Note that some other known multiple access methods such as FDMA, CDMA and OFDMA leveraging different communication resources may be able to result in latency reduction.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3118928, IEEE Internet of Things Journal

to node j than the transmit node i, and thus, in the rest of this article, we design T-TDMA to control the interference level to be negligibly small, e.g., $I_{ij} \ll P_n$ for all $j \in \mathcal{N}$, compromising the latency performance. For such case, we can approximate the SINR (2) as

$$\mathrm{SINR}_{ij} \approx 10^{-\frac{\mathrm{PL}_{dB}(R_0)}{10}} \frac{|H_{ij}|^2 P_t}{P_n} \left(\frac{R_0}{R_{ij}}\right)^{\eta} \tag{4}$$

where (4) is semantically represents the signal-to-noise ratio (SNR).

The absolute time is globally synchronized periodically with GPS [38], [39] and split into time slots of fixed time interval such that

$$\tau_o = \frac{M}{B\log(1+\rho)} \text{ seconds,} \tag{5}$$

where M is the maximum size of message sent by nodes during the consensus process in bits, ρ is the target signal-to-interference-plus-noise ratio (SINR) of each transmission, and B (Hz) denotes the bandwidth utilized for transmission. An SINR outage occurs if SINR is below the given threshold ρ. Since $|H_{ij}|^2$ is exponentially distributed, the SINR outage probability ϵ_{ij} is given as

$$\begin{aligned} \epsilon_{ij} &= \Pr[\mathrm{SINR}_{ij} < \rho] && (6)\\ &= \Pr\left[|H_{ij}|^2 < 10^{\frac{\mathrm{PL}_{dB}(R_0)}{10}} \rho \frac{P_n}{P_t}\left(\frac{R_{ij}}{R_0}\right)^{\eta}\right] && (7)\\ &= 1-\exp\left(-10^{\frac{\mathrm{PL}_{dB}(R_0)}{10}} \rho \frac{P_n}{P_t}\left(\frac{R_{ij}}{R_0}\right)^{\eta}\right). && (8) \end{aligned}$$

In the rest of this paper, we analyzed the latency performance utilizing the SINR outage probability (8) derived from statistical channel model. From an operation point of view in the actual channel condition, if an outage occurs due to a poor condition of the actual channel, retransmissions are carried out via the type-I hybrid automatic repeat request (H-ARQ) until the lost messages are successfully delivered. These transmission failures and retransmissions are taken into account in the performance analysis and the protocol design in a statistical sense.

C. Individual-Broadcasting Time Duration

Every communication method considered in this paper is built upon broadcasting leveraged at each node. We assume that every node, as a transmitter, has its *target communication nodes*, which literally are the communication targets of a transmitter node. For the efficient use of radio resources, we allocate $x_i\tau_o$ seconds of *individual-broadcast time duration* for a message broadcast by an individual node $i \in \mathcal{N}$ as a transmitter. We use the term 'individual' to distinguish from the cooperative-broadcast (CB) which will appear in the following section. Let X_i be a random variable denoting the number of time slots required for message transmission to all target communication nodes of the node i. In the individual broadcasting, we suppose that x_i is determined to be minimum x such that

$$\Pr[X_i \le x] \ge \zeta \tag{9}$$

for some target broadcast success probability $0 \le \zeta < 1$. Then we can obtain the following proposition.

Proposition 1. *The required number of time slot x_i to broadcast a message from i to its n target nodes with probability $0 \le \zeta < 1$ is*

$$x_i = g(n, \epsilon_{i\,\max}), \tag{10}$$

where the function $g(a,b)$ is defined as $g(a,b) = \left\lceil \log\left(1-\zeta^{\frac{1}{a}}\right)/\log b \right\rceil$, for $a \in \mathbb{N}$ and $b \in (0,1]$ and $\epsilon_{i\,\max} = 1-\exp\left(-10^{\frac{\mathrm{PL}_{dB}(R_0)}{10}} \rho \frac{P_n}{P_t}\left(\frac{\max_{j\in\mathcal{J}} R_{ij}}{R_0}\right)^{\eta}\right)$ is the SINR outage probability between node i and the node located maximum distance apart from node i among the target communication node set $\mathcal{J}$.

Proof: Let X_{ij} be a random variable which denotes the number of time slots required to deliver a message from source node i to destination node $j \in \mathcal{J}$ via broadcast. In this case, the message is delivered in a single hop. Since the transmission failures might occur with certain probability due to channel fading and the message is sent repeatedly until successful delivery, the random variable X_{ij} follows a geometric distribution with p.d.f.

$$\Pr[X_{ij} = x] = \begin{cases} \epsilon_{ij}^{x-1}\left(1-\epsilon_{ij}\right), & \forall x \ge 1 \\ 0, & \text{elsewhere,} \end{cases} \tag{11}$$

where $\epsilon_{ij} = \Pr\left[\mathrm{SINR}_{ij} < \rho\right]$ is the SINR outage probability (7) with transmit power P_t. Then the dissemination outage probability can be upper bounded as

$$\begin{aligned} \Pr\left[\max_j X_{ij} \le x\right] &= \prod_{j\in\mathcal{J}} \Pr\left[X_{ij} \le x\right] && (12)\\ &= \prod_{j\in\mathcal{J}} \left(1-\epsilon_{ij}^{x}\right) && (13)\\ &\ge \left(1-\epsilon_{i\,\max}^{x}\right)^{n}. && (14) \end{aligned}$$

The equality (12) holds from the independence of the channels between the transmit node and the receiving nodes, and the inequality (14) holds from the fact that the SINR outage probability is the largest when the node j is located farthest apart from the node i among all the receiving nodes from (7). Since the required number of time slot x_i is determined to be minimum x which satisfy the broadcast success probability larger than ζ, we can use the lower bound of success probability (14) and the required number of time slot x_i becomes

$$x_i = \left\lceil \frac{\log\left(1-\zeta^{\frac{1}{n}}\right)}{\log \epsilon_{i\,\max}} \right\rceil, \tag{15}$$

from (14) is equal to ζ. ∎

III. Gossip-Broadcasting based Referendum Consensus

In the conventional RC, the messages were disseminated via either gossiping or broadcasting. The gossiping method is a multi-hop dissemination protocol, in which the nodes transmit messages to only adjacent nodes to disseminate message to the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3118928, IEEE Internet of Things Journal

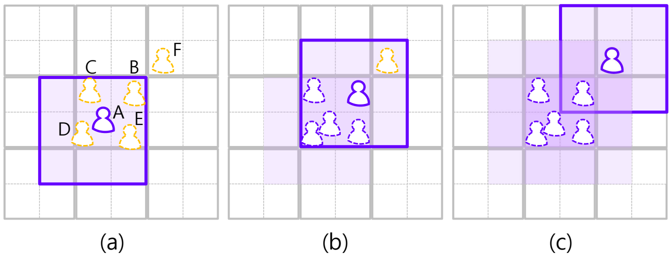


Fig. 1: An illustration of the gossip-broadcasting message dissemination method example stated in Sec. III-A.

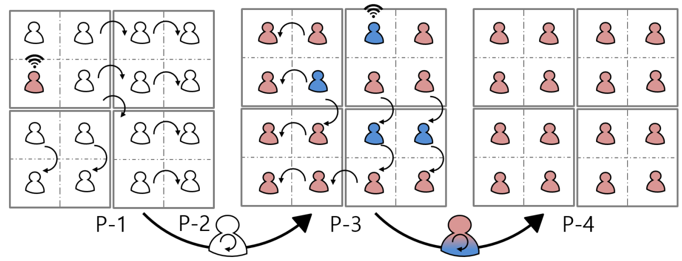


Fig. 2: An illustration of the gossip-broadcasting based referendum consensus (GB-RC) process: **P-1** proposer disseminates proposal message via the GB, **P-2** each node starts local validation, **P-3**: after the local validation, each validator node disseminate its commit message via the GB, **P-4**: through the global validation, every node reaches consensus.

whole network, and broadcasting method transmit messages to whole network through at single-hop. By combining the both methods, this section proposes a novel *gossip-broadcasting based referendum consensus (GB-RC)* that leverages a message dissemination method termed *gossip-broadcasting (GB)* to enjoy the benefit of each of them.

A. Gossip-Broadcasting Dissemination

The GB is a method that essentially disseminates messages locally by broadcasting and globally by gossiping. Basically, during the GB, each node only leverages individual broadcasting, but the messages are disseminated by multi-hop, which resembles the wireless gossiping. Refer to the following illustrative example that explains the GB method (see Fig.1).

Consider a network that consists of 9 local networks, and a node, say Alice, is in the local network k and she is disseminating a message M_A to the rest of the nodes in the global network. To do this, all local networks are virtually divided into 4 regions as shown in Fig. 1. In each region, a single relay is randomly chosen among the nodes in the region through the predefined leader election process. Then, the message dissemination with GB is done with the steps described below.

(a) First, Alice broadcasts M_A to all the target communication nodes. In the GB, we let the target communication nodes be in the 9 virtually divided regions, including the one which the transmitter is in and its 8 neighboring regions (the colored regions in Fig. 1). Thus, 4 pre-elected relays in the local network k, i.e., Bob, Charlie David, and Eve, can receive M_A.
(b) Next, Bob, Charlie David, and Eve broadcast M_A by taking turns with the Round-Robin approach. The broadcasting order of the relay nodes is also decided by a consensus process between the nodes [31]. When they have finished broadcasting successfully, at least one of the relays in each neighboring local network has received M_A.
(c) Let Frank be the relay in one of the neighboring local networks who has received M_A from David. Then Frank broadcasts M_A and consequently all nodes in the Frank's local network will also receive M_A. The nodes in the other neighboring local networks, which are the target communication nodes of Frank, will also receive M_A in the similar way as Frank did from David.

In the example above, we only considered a global network with 9 local networks, however, by applying this method in a larger network with more than 9 local networks, we can still have successful message dissemination. Meanwhile, in the GB, we divide local networks into 4 regions assuming that the target communication nodes of each transmitter to be all nodes within its divided regions and 8 neighboring regions. The reason for this is to avoid further interference issues that might happen even though we use TDMA scheme. However, if we consider a larger-scale time division multiple access scheme, by compromising communication efficiency, the target communication range can become larger and a single relay in one local network can be enough to operate the GB.

B. Operational structure of GB-RC

Here, we state the operational structure of the GB-RC that incorporates 4 operational phases. Refer to the brief illustration explaining the operation of GB-RC provided in Fig. 2.

Phase 1: *(Action proposal)* Let node $p \in \mathcal{N}$ in a local network $k \in \mathcal{K}$ be the proposer node and the proposed action be A_p. The other nodes in the global network $\mathcal{V}_p = \mathcal{N} \setminus \{p\}$ participate in the consensus process as validator nodes. In this phase, p initiates the GB-RC process by broadcasting a signed action proposal message M_p[4]. The message sent by p for A_p is denoted by $\mathsf{M}_p^{A_p}$ and consists of

$$\mathsf{M}_p^{A_p} = [A_p, T(A_p), O(\mathcal{V}_p)] \tag{16}$$

where $T(A_p)$ is the absolute time stamp describing when the action has been proposed, and $O(\mathcal{V}_p)$ is a randomly determined order of the validators. The validators will commit their messages based on this order in Phase 3. During the proposal phase, $\mathsf{M}_p^{A_p}$ is passed to all validators via GB dissemination.

Phase 2: *(Local Validation)* After each node receives the message $\mathsf{M}_p^{A_p}$, it starts to validate A_p. The validation at each validator node is carried out with its own distributed ledger (validation results are illustrated as node color in Fig.2). The local validity at the validator $v \in \mathcal{V}_p$ about A_p is denoted by $V(A_p)_v$.

[4]We assume that the messages contain public-key signatures [40]. Throughout the article, any message M signed by node i is denoted by M_i. Thus the encrypted signature of M from i is included in M_i.

Phase 3: *(Commit)* After the local validations are completed, the validator nodes take turn to disseminate the commit messages. The committing order is specified in $O(\mathcal{V}_p)$. Note that the validators also leverage the GB dissemination to send the commit messages to the other nodes. The commit message of v about A_p consists of

$$\mathsf{M}_v^{A_p} = [V(A_p)_v, T(A_p)]. \tag{17}$$

Phase 4: *(Global Validation & Action Ordering).* After each validator collects more than $N-F$ local validity, it determines the global validity of the proposed action. Note F is the number of faulty nodes in the global network. Essentially, GB-RC follows the majority rule to determine the global validity and it is known that the system is resilient against the faulty nodes if the condition $N > 3F$ holds [11]. On the other hand, if there are more than tolerable faulty nodes in the network, i.e., $F > N/3$, the network fails to reach consensus. If the network successfully reaches global consensus, all the nodes in the network share the same validity. The order of the valid actions are organized based on the timestamp $T(A_P)$. Note that due to the reorganization of the valid actions based on the timestamp, there might be some contradictory actions that might violate the causal relation of the actions. Such troublesome actions are unaccepted and announced to be retried later on or discarded.

C. Consensus Latency of GB-RC

The *consensus latency* is defined as the time elapsed from the action proposal to the global consensus which includes global validation and action ordering. Compare to the hashing operation-based distributed consensus mechanisms including [22], [41], [42], computational complexity of BFT-based distributed consensus at each node is low, which makes the consensus latency that comes from the computing operation is relatively smaller than that of hashing operation-based distributed consensus schemes. Instead, the number of communication rounds and scalability is the main issue that determines consensus latency in BFT-based consensus [10], [11] and thus we assume that the local computing time is negligibly small compared to the communication delay during the consensus process. Moreover, by neglecting the computational term in the latency, we can more focus on the impact of wireless channel conditions and environments on the consensus latency. Thus, the latency of the GB-RC is determined by the time for proposing and committing, i.e., Phase 1 and Phase 3. Accordingly, a single round of GB-RC consists of N independent message dissemination processes. In addition, different from the conventional RC mechanism, consider the following two properties of the GB method in the GB-RC latency analysis.

- The dissemination time depends on the number of required relay nodes in the shortest paths to the target nodes. The shortest path to the most distant local network depends on transmitter's position.
- The number of target communication nodes affects the individual-broadcasting time duration as mentioned in Sec II-C. We focus on the case when each local network is virtually divided into square-shaped region with the same size, as mentioned earlier and shown in Fig. 2.

Then, the upper bound of the GB-RC latency with a target broadcast success probability larger than or equal to ζ, for $0 \leq \zeta < 1$ can be written as follows.

Theorem 1. *The upper bound of the GB-RC latency is*

$$L_{\textit{GB-RC}} \leq TN\left(4|\bar{\mathcal{K}}_{\max}| - 2\right) g\left(\frac{9D}{4} - 1, \epsilon_{\max}\right)\tau_o \tag{18}$$

where $\epsilon_{\max} = 1 - \exp\left(-10^{\frac{PL_{dB}(R_0)}{10}}\rho\frac{P_n}{P_t}\left(\frac{R(\sqrt{2D}-\sqrt{2})}{R_0}\right)^{\eta}\right)$, *and*

$$|\bar{\mathcal{K}}_{\max}| = \begin{cases} \frac{5\sqrt{K}}{6} - \frac{1}{3\sqrt{K}} + \frac{1}{2}, & \sqrt{K} \textit{ is even,} \\ \frac{5\sqrt{K}}{6} - \frac{11}{6\sqrt{K}} + \frac{3}{2K} + \frac{1}{2}, & \sqrt{K} \textit{ is odd.} \end{cases} \tag{19}$$

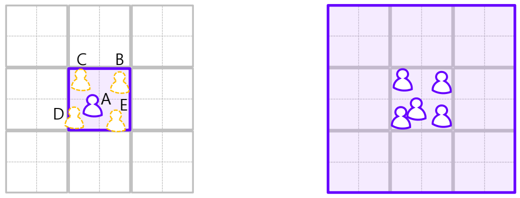
Proof: The proof is provided in Appendix A. ■

D. Improving Scalability

By compromising the reliability of the consensus mechanism, we can further improve the scalability of the GB-RC. Suppose that the proposed action is a transaction between 2 nodes, similar to the Blockchain and other crypto-currencies [22], [43], and it only affects the local networks in which the nodes are located. Then we possibly reduce the number of validators, by only allowing the nodes in the corresponding local network to participate in the consensus process. The other nodes, which did not participate in the consensus process, update their ledgers through periodic global synchronization across the local networks. The main advantage of the improved GB-RC over the original GB-RC is its scalability due to the following features.

- The number of validators is independent of the global network. It only depends on the number of nodes in the local networks, which the two transactors are located. Therefore the latency for committing the proposed action will dramatically decrease, thereby decreasing the overall consensus latency significantly compared to GB-RC.
- If the local networks that are involved in the consensus are located in short distance, the required time to finish the dissemination will be further reduced. In contrast, each node's message should be disseminated to the most distant node in the global network in GB-RC.
- During the message dissemination via GB, the relays do not require to broadcast to every nodes in adjacent regions. The relay nodes only need to consider the next relays as their target communication nodes. Therefore the required retransmissions can be reduced.

As mentioned earlier, while the improved GB-RC has a great advantage of reducing latency and improving scalability, the resiliency should be compromised. As the number of validators participating in the consensus decreases, the system may become unstable with only a small number of faulty nodes. Therefore, the improved GB-RC should be used carefully considering such trade-off relationship.



(a) (b)

Fig. 3: An illustration of the cooperative-broadcasting message dissemination method example stated in Sec. IV-A.

IV. Cooperative-Broadcasting based Electoral-College Consensus (CB-EC)

In the GB-RC, the actions proposed by the nodes are validated by all nodes, where each node gets a vote that has equal value. The problem is that even though the GB dissemination method achieves communication-efficiency, the fundamental structure of referendum consensus bogs down the scalability of the GB-RC. The proposed *cooperative-broadcast based electoral-college consensus (CB-EC)*, which is a kind of representative consensus, overcomes this limitation by leveraging the notion of presidential election of the U.S. In CB-EC, the voting value of each local network is determined by the number of nodes in local network. The winner validity (i.e., valid or not) of each local network, takes the opportunity to vote in the global consensus with the assigned vote value. To this end, the CB-EC adopts a *cooperative-broadcast* (CB) dissemination method, which is elaborated in the following.

A. Cooperative-Broadcasting Dissemination

Instead of leveraging the notion of gossiping as we did in the GB dissemination method, the CB dissemination method leverages a synchronized and simultaneous broadcasting method that are done cooperatively by all the nodes in a local network [13], [44]. All nodes in the local network transmit the same message to every other nodes in global network in a common time slot. Refer to the following illustrative example that explains the CB method (see Fig.3).

Consider a network consisting of 9 local networks. Alice, who is in the local network $k \in \{1,\dots,9\}$, is trying to disseminate a message M_A to all the other nodes in global network. Then, the message dissemination is done with following steps described below.

(a) Say Bob, Charlie and David are in the same local network k as Alice. Alice first shares the message M_A to Bob, Charlie, and David via individual broadcasting.
(b) After Bob, Charlie and David receive M_A, Alice, Bob, Charlie, and David cooperatively broadcast M_A to the other local networks.

The process of the CB dissemination method does not require the use of relay for the *reliable message dissemination*. Instead, the key concept of this method is to leverage the diversity gain from multiple nodes' synchronous and cooperative transmissions as if they are a virtual multi-antenna transmission. Note that the nodes can synchronize the transmission by referring to the timestamp provided in M_A and the multipath delay can be neglected since the broadcast signal is a narrowband signal.

Meanwhile, by expanding the individual-broadcast time duration proposed in Sec. II-C, we analyze the *cooperative-broadcast (CB) time duration*. In the CB dissemination method, the target communication nodes for the CB is all the nodes in the global network. Different from the individual-broadcast, suppose allocating $\bar{x}_{(k)}\tau_o$ seconds of CB time duration for a local network $k \in \mathcal{K}$. Let $\bar{X}_{(k)}$ be a random variable denoting the number of time slots required for message transmission to all the other nodes. In the CB, we suppose that $\bar{x}_{(k)}$ is determined to be minimum x such that

$$\Pr[\bar{X}_{(k)} \leq x] \geq \zeta \tag{20}$$

for some target cooperative broadcast success probability $0 \leq \zeta < 1$. Then we can obtain following proposition.

Proposition 2. *Required number of time slot $\bar{x}_{(k)}$ to transmit a message from nodes in local network $k \in \mathcal{K}$ to other n nodes by cooperative-broadcast with probability $0 \leq \zeta < 1$ is*

$$\bar{x}_{(k)} = 2|\mathcal{N}_k| g(n, \epsilon_{(k)\,\max}) \tag{21}$$

where $\epsilon_{(k)\,\max}$ is (33), *which describes the SINR outage probability by the cooperative transmission between the local network k and node located maximum distance apart from node i among the target communication node set $\mathcal{J}$.*

Proof: If all nodes in the local network k transmit a message simultaneously using the pre-designed distributed orthogonal space-time code for $|\mathcal{N}|_k$ transmit nodes [13], then receiver signal-to-interference-plus-noise ratio (SINR) at node j can be obtained as

$$\begin{aligned} \mathrm{SINR}_{(k)j} &= 10^{-\frac{\mathrm{PL}_{dB}(R_0)}{10}} \sum_{i\in\mathcal{N}_k} \frac{|H_{ij}|^2 P_t}{P_n + I_{ij}} \left(\frac{R_0}{R_{ij}}\right)^{\eta}. \qquad (22) \\ &\approx 10^{-\frac{\mathrm{PL}_{dB}(R_0)}{10}} \sum_{i\in\mathcal{N}_k} \frac{|H_{ij}|^2 P_t}{P_n} \left(\frac{R_0}{R_{ij}}\right)^{\eta}, \qquad (23) \end{aligned}$$

with well designed T-TDMA. Applying this equation to (7), we can rewrite SINR outage probability of cooperative broadcasting, $\epsilon_{(k)j}$, as

$$\begin{aligned} \epsilon_{(k)j} &= \Pr[\mathrm{SINR}_{(k)j} < \rho] \\ &= \Pr\left[\sum_{i\in\mathcal{N}_k} |H_{ij}|^2 \left(\frac{R_0}{R_{ij}}\right)^{\eta} < 10^{\frac{\mathrm{PL}_{dB}(R_0)}{10}} \rho \frac{P_n}{P_t}\right] \end{aligned} \tag{24}$$

Let $\bar{X}_{(k)j}$ be a random variable which denotes the number of time slots required to deliver a message from the local network k to node j via cooperative-broadcast. Since the messages are sent repeatedly until successful, the random variable $\bar{X}_{(k)j}$ follows the geometric distribution with p.d.f.

$$\Pr\left[\bar{X}_{(k)j} = x\right] = \begin{cases} \epsilon_{(k)j}^{x/c-1}\left(1-\epsilon_{(k)j}\right), & \forall x \geq 1 \\ 0, & \text{elsewhere.} \end{cases} \tag{25}$$

where c is the time scaling constant that occurs due to rate reduction by distributed orthogonal space-time coding

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3118928, IEEE Internet of Things Journal

[45]. Then the dissemination outage probability can be upper bounded as

$$\Pr\left[\max_j \bar{X}_{(k)j} \le x\right] = \prod_{j\in\mathcal{J}} \Pr\left[\bar{X}_{(k)j} \le x\right] \quad (26)$$

$$= \prod_{j\in\mathcal{J}} \left(1 - \epsilon_{(k)j}^{x/c}\right) \quad (27)$$

$$\ge \left(1 - \epsilon_{(k)\max}^{x/c}\right)^n \quad (28)$$

$$\ge \left(1 - \epsilon_{(k)\max}^{x/2|\mathcal{N}_k|}\right)^n, \quad (29)$$

where $\epsilon_{(k)\max}$ is the SINR outage probability between the local network k and the node located maximum distance apart from k. The equality (26) holds from the independence of the channels between the transmit node and the receiving nodes, and the inequality (28) comes from the fact that the SINR outage probability is the largest when node j is located farthest apart from local network k among all the receiving nodes from (24). The inequality (29) holds because the required time scale c is upper bounded by $2|\mathcal{N}_k|$ [45]. Assuming that every node in $\mathcal{N}_k$ is located in the center of k, which is a tight approximation for distant nodes from k, we can approximate $\epsilon_{(k)\max}$ as

$$\epsilon_{(k)\max} = \Pr[\text{SINR}_{(k)\max} < \rho] \quad (30)$$

$$= \Pr\left[\sum_{i\in\mathcal{N}_k} |H_{i\max}|^2 \left(\frac{R_0}{R_{i\max}}\right)^\eta < 10^{\frac{\text{PL}_{dB}(R_0)}{10}} \rho \frac{P_n}{P_t}\right] \quad (31)$$

$$\approx \Pr\left[\sum_{i\in\mathcal{N}_k} |H_{i\max}|^2 < 10^{\frac{\text{PL}_{dB}(R_0)}{10}} \rho \frac{P_n}{P_t} \left(\frac{R_{(k)\max}}{R_0}\right)^\eta\right] \quad (32)$$

where $R_{i,\max}$ is the distance between node i to the most distant node, which can be approximated to the distance from center of the local network k to the most distant node, $R_{(k)\max}$, from the assumption. Moreover $H_{i,\max}$ is the channel gain from node i to the most distant node so we can obtain (33) since $\sum_{i\in\mathcal{N}_k} |H_{i\max}|^2$ follows the Erlang distribution as $\sum_{i\in\mathcal{N}_k} |H_{i\max}|^2 \sim \text{Erlang}(|\mathcal{N}_k|; 1)$. Since the required number of time slots $\bar{x}_{(k)}$ is determined to be the minimum x such that the cooperative-broadcast success probability becomes larger than ζ. Hence, by using the lower bound of the success probability (29), the required number of time slot $\bar{x}_{(k)}$ becomes

$$\bar{x}_{(k)} = 2|\mathcal{N}_k| \left\lceil \frac{\log\left(1 - \zeta^{\frac{1}{n}}\right)}{\log \epsilon_{(k)\max}} \right\rceil, \quad (34)$$

by equating (29) to ζ. ∎

B. Operational Structure of CB-EC

The whole process of CB-EC is composed of the following five operational phases. Refer to the brief illustration of the process in Fig.4.

Phase 1: *(Action proposal)* The proposer p proposes an action A_p to all the validator nodes $\mathcal{V}_p = \mathcal{N}\backslash p$ in the global network. The proposer initiates the CB-EC by individually broadcasting a signed action proposal message consisting of

$$\mathsf{M}_p^{A_p} = [A_p, T(A_p), O_1(\mathcal{V}_p), O_2(\mathcal{V}_p)] \quad (35)$$

where $T(A_p)$ is the absolute timestamp when node p proposed A_p, $O_1(\mathcal{V}_p)$ is the committing order of the validator nodes in Phases 3, and $O_2(\mathcal{V}_p)$ is the committing order of the local networks in Phase 4. The message M_p will be disseminated to the global network through the CB method.

Phase 2: *(Local Action Validation)* When all the nodes have received the message, they start doing local validation to validate the proposed action A_p. The local validity at validator $v \in \mathcal{V}_p$ for A_p is denoted by $V(A_p)_v$.

Phase 3: *(Local Commit & Semi-Global Validation)* After the local validation ends, at each local network, the validator nodes individually broadcast their commit messages to the other nodes in their local network by taking turns according to the committing order specified in $O_1(\mathcal{V}_p)$ which was informed by the proposer in Phase 1. The local commit phase is executed in parallel in each local network, as interference from neighboring local networks can be ignored by the previously mentioned T-TDMA scheme. The commit message of node v is denoted by

$$\mathsf{M}_v^{A_p} = [V(A_p)_v, T(A_p)]. \quad (36)$$

When each validator node collects more than $|\mathcal{N}_k| - F_k$ commit messages, it determines the *semi-global validity* of the proposed action by the majority rule, where F_k is the number of faulty nodes in each local network k, $\forall k \in \mathcal{K}$. The consensus of local network k is resilient against faulty node if the condition $|\mathcal{N}_k| > 3F_k$ holds. Once a local network reaches a semi-global consensus, all the member nodes will have the same validity denoted by $V(A_p)_k$, $\forall k \in \mathcal{K}$.

Phase 4: *(Global Validation & Action Ordering)* If the semi-global validation is finished in every local network, the semi-globally validated messages are disseminated to the other local networks by taking turns according to the committing order specified in $O_2(\mathcal{V}_p)$. When committing, the validators in the local network k transmits their semi-validated commit message,

$$\mathsf{M}_k^{A_p} = [V(A_p)_k, T(A_p)], \quad (37)$$

to the other local networks leveraging the CB dissemination method. Among the nodes in local network k, a fixed number of nodes are selected randomly and their signatures are used as the signature of local network k.

$$\epsilon_{(k)\max} = 1 - \sum_{n=0}^{|\mathcal{N}_k|-1} \frac{1}{n!} \exp\left(-10^{\frac{\text{PL}_{dB}(R_0)}{10}} \rho \frac{P_n}{P_t} \left(\frac{R_{(k)\max}}{R_0}\right)^\eta\right) \left(10^{\frac{\text{PL}_{dB}(R_0)}{10}} \rho \frac{P_n}{P_t} \left(\frac{R_{(k)\max}}{R_0}\right)^\eta\right)^n. \quad (33)$$

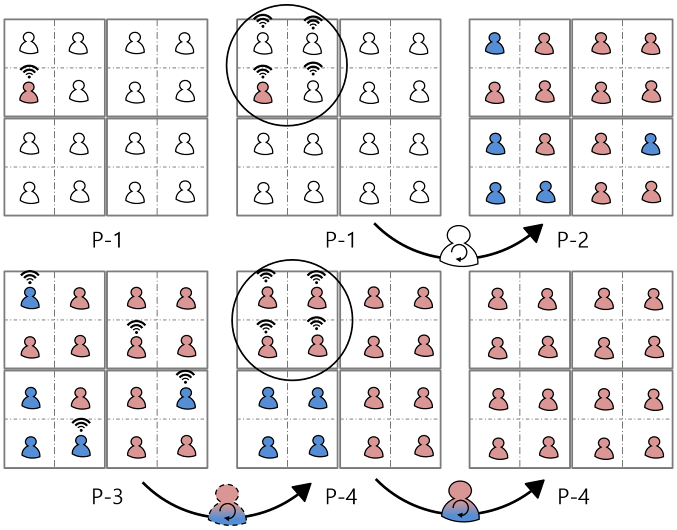


Fig. 4: An illustration of the cooperative-broadcasting based electoral-college consensus (CB-EC): **P-1** the proposer disseminates $\mathsf{M}_p^{A_p}$ by the CB, **P-2** each node starts local validation, **P-3** after the local validation, each node broadcasts its commit message $\mathsf{M}_v^{A_p}$ to the nodes in its local network, **P-4** after the semi-global validation, each local network disseminates $\mathsf{M}_k^{A_p}$ by the CB and through the global validation, every node reaches a consensus.

After all validators in the global network receive the commit messages from all the other local networks, they determine the global validity. Note the local networks which failed to reach the correct consensus in Phase 3, named faulty local networks, will send incorrect messages to the others, thereby hindering the global validation. The system is resilient against faulty local networks if

$$N > 3\sum_{k\in\mathcal{K}} |\mathcal{N}_k|\mathbb{1}\{|\mathcal{N}_k| \leq 3F_k\} \tag{38}$$

holds where F_k is the number of faulty nodes in local network k. For the globally reached consensus, the order of the valid actions are organized based on the time stamp $T(A_P)$. Note that due to the reorganization of the valid actions based on the time stamp, troublesome actions are unaccepted and announced to be retried later on or discarded. Since the validation is based on the local network level majority rule, the reliability of the CB-EC compromises resiliency. For this reason, we study the resiliency of the CB-EC against faulty nodes in the following subsection.

C. Resiliency of CB-EC

For given F faulty nodes in the global network, the GB-RC is resilient if it satisfies $N > 3F$. Likewise, the CB-EC is resilient if the total number of nodes satisfies (38) for given number of faulty node $F = \sum_{k\in\mathcal{K}} F_k$. According to this, the sufficient condition on the number of faulty nodes that we can guarantee CB-EC to be resilient against the faulty nodes is given by $F < \lceil N/9 \rceil$, which is independent of the number of nodes in each local network.

Since a local network's faultiness is determined by the distribution of faulty nodes in the global network, the resiliency of the CB-EC can be described in a stochastic manner. For mathematical tractability, we focus on the case that the number of nodes is equal in all local networks. Then, from the fact that the CB-EC is resilient if it satisfies $K > 3\tilde{F}$ against $\tilde{F}$ faulty local networks, the stochastic α-resiliency is defined as follows.

Definition 1. *(Stochastic α-resiliency).* For a fixed number of faulty nodes F randomly distributed over a network, the CB-EC is *α-resilient* if

$$\Pr[K > 3\tilde{F}] \geq \alpha, \tag{39}$$

for a resiliency probability α where $0 < \alpha \leq 1$ while $\tilde{F}$ is the number of faulty local networks among K local networks comprising the global network.

Next, we characterize the maximum number of faulty nodes F that can ensure α-resiliency of the CB-EC. To begin with, consider the perfect resiliency case, i.e., $\alpha = 1$. Then the number of faulty local networks $\tilde{F}$ should satisfy $K > 3\tilde{F}$ no matter how the faulty nodes are distributed in the global network. Thus, if more than $K - \lceil K/3 \rceil$ local networks yield semi-global validity, it is possible to reach global consensus. Let D be the number of nodes in each local network, and if there are more than $\lceil D/3 \rceil - 1$ faulty nodes in the local network, it is not resilient against the faulty nodes and local consensus in the local network is faulty. Note that if the number of faulty nodes in the global network is less than $F_* = \lceil D/3 \rceil \lceil K/3 \rceil$, i.e., $F < F_*$, the CB-EC ensures the perfect resiliency. In other words, if there are more than or equal to F_* faulty nodes in the global network, i.e., $F \geq F_*$, they may hinder perfect resiliency of the CB-EC. Compared to the sufficient condition $F < \lceil N/9 \rceil$ for the general case, $F < F_*$ is the relaxed condition due to the assumption that every local network has the equal number of nodes.

Now we focus on the cases for $\alpha < 1$, which corresponds to the case when $F \geq F_*$. To begin with, notice that if the CB-EC results in faulty consensus, there should be at least $\lceil K/3 \rceil$ faulty local networks. This means that the faulty consensus happens in CB-EC if arbitrary $\lceil K/3 \rceil$ local networks include arbitrary $\lceil D/3 \rceil$ faulty nodes, no matter how the remaining $F - F_*$ faulty nodes distributed over the global network. Thus by defining $\mathcal{E}_{\text{faulty}}$ as the set of all faulty consensus events, and $\mathcal{E}_i$ as the sets of events of distributing arbitrarily chosen F_* faulty nodes to arbitrarily chosen $\lceil K/3 \rceil$ local networks so that each has $\lceil D/3 \rceil$ faulty nodes for all $i \in \left\{1, \ldots, \binom{F}{F_*}\right\}$. Thus, from the statement above, we have

$$\mathcal{E}_{\text{faulty}} = \mathcal{E}_1 \cup \mathcal{E}_2 \cup \cdots \cup \mathcal{E}_{\binom{F}{F_*}}. \tag{40}$$

As a result,

$$\begin{aligned}
\alpha &= 1 - \mathbf{P}[\mathcal{E}_{\text{faulty}}] && (41)\\
&= 1 - \mathbf{P}\left[\mathcal{E}_1 \cup \mathcal{E}_2 \cup \cdots \cup \mathcal{E}_{\binom{F}{F_*}}\right] && (42)\\
&\geq 1 - \left(\mathbf{P}[\mathcal{E}_1] + \mathbf{P}[\mathcal{E}_2] + \cdots + \mathbf{P}\left[\mathcal{E}_{\binom{F}{F_*}}\right]\right) && (43)\\
&= 1 - \binom{F}{F_*}\mathbf{P}[\mathcal{E}] && (44)
\end{aligned}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3118928, IEEE Internet of Things Journal

$$= 1 - \binom{F}{F_*}\frac{|\mathcal{E}|}{N!}, \tag{45}$$

where we use a general term $\mathcal{E}$ to denote any set of events among $\mathcal{E}_1, \ldots, \mathcal{E}_{\binom{F}{F_*}}$, since $\mathbf{P}[\mathcal{E}_1] = \cdots = \mathbf{P}\left[\mathcal{E}_{\binom{F}{F_*}}\right]$ and $|\mathcal{E}_1| = \cdots = \left|\mathcal{E}_{\binom{F}{F_*}}\right|$ hold without loss of generality. (42) is obtained from the equation (40), and the inequality in (43) holds by the union bound. (45) is from the definition of $\mathbf{P}[\mathcal{E}]$. Furthermore, note that

$$|\mathcal{E}| = (N - F_*)!(F_*)!\binom{K}{\lceil K/3\rceil}\binom{D}{\lceil D/3\rceil}^{\lceil K/3\rceil}, \tag{46}$$

where $\binom{K}{\lceil K/3\rceil}$ chooses $\lceil K/3\rceil$ local networks out of K, $\binom{D}{\lceil D/3\rceil}^{\lceil K/3\rceil}$ chooses $\lceil D/3\rceil$ faulty nodes out of D nodes from all $\lceil K/3\rceil$ chosen local networks. The terms $F_*!$ and $(N - F_*)!$ respectively consider all possible placement of F_* faulty nodes and the other $N - F_*$ nodes in the network. Overall, we have the following proposition.

Proposition 3. *Depending on the condition of the number of faulty node F, the CB-EC is α-resilient to the faulty nodes with*

$$\alpha = 1, \qquad F < F_*, \tag{47}$$

$$\alpha \geq \max\left(1 - \frac{\binom{F}{F_*}|\mathcal{E}|}{N!}, 0\right), \qquad F \geq F_*, \tag{48}$$

where $F_* = \lceil D/3\rceil \lceil K/3\rceil$ and $|\mathcal{E}|$ is (46)*.*

The equality in (48) holds when $F = F_*$, since the equality holds in (43). For given α, the maximum allowed number of faulty nodes F that CB-EC can achieve the α-resiliency can be readily obtained by simple algorithms such as binary search between F_* and N, since $1-\binom{F}{F_*}|\mathcal{E}|/N!$ is a decreasing function of F.

D. Consensus Latency of CB-EC

The latency of the CB-EC with the target individual/cooperative broadcast success probability ζ is determined by the time for proposing, local committing, and global committing. The computing time is neglected as we did for the latency of GB-RC. Since the CB-EC latency depends on the location of the proposal node, we will analyze the worst case of of the CB-EC latency for a fair comparison with GB-RC. As a result, we obtain an upper bound of the CB-EC latency as follows.

Theorem 2. *An upper bound of the CB-EC latency is*

$$L_{\text{CB-EC}} \leq T\big((D+1)g(D-1, \epsilon_{\max}) + 2D(K+1)g(N-D, \bar{\epsilon}_{\max})\big)\tau_o \tag{49}$$

where $\epsilon_{\max} = 1 - \exp\left(-10^{\frac{PL_{dB}(R_0)}{10}}\rho\frac{P_n}{P_t}\left(\frac{R(\sqrt{2D}-\sqrt{2})}{R_0}\right)^{\eta}\right)$ *and*

$$\bar{\epsilon}_{\max} = 1 - \sum_{n=0}^{D-1}\frac{1}{n!}a^n\exp(-a) \tag{50}$$

while $a = 10^{\frac{PL_{dB}(R_0)}{10}}\rho\frac{P_n}{P_t}\left(\frac{\sqrt{2}R(2\sqrt{N}-\sqrt{D}-1)}{2R_0}\right)^{\eta}$.

Proof: The proof is provided in Appendix B. ■

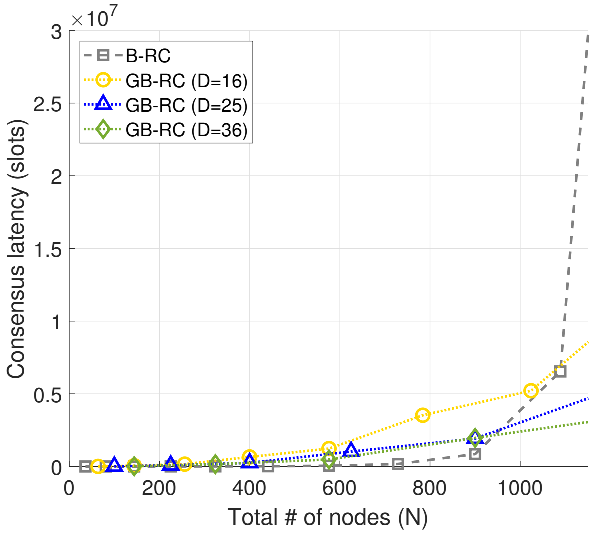
V. Numerical results

In this section, we numerically evaluate the performance of the GB-RC and the CB-EC, and validate the analytic results obtained in the previous sections. Taking into account of the communications between IoT devices, e.g., devices equipped with Bluetooth-based transceivers, we fix the transmit power at each node for broadcast transmissions as $P_t = 100$ mW, and the noise power as $P_n = 10^{-10}$ mW. In addition, we set the distance between the two neighboring nodes to be $R = 10$ meters, and TDMA parameter $T = K$ is chosen to make sure the interference is negligibly small to have the approximation (4) regardless of the other factors that affects interference. In practice, T can be designed more elaborately, knowing the information about other factors. The target dissemination success probability is $\zeta = 0.999$ and the target SINR threshold is assumed to be $\rho = 10$ dB. We also fix the path loss exponent as $\eta = 3$, which usually ranges between $2.7 - 3.5$ for urban outdoor scenarios and between $1.6 - 3.3$ for indoor scenarios [46]. Moreover, from the Friis equation, we assume $\text{PL}_{\text{dB}}(R_0) = 20\log_{10}\frac{\lambda}{4\pi R_0}$ and for the simulation we fix $R_0 = 1$ meter and $\lambda = 0.125$ meters, from the industrial, scientific and medical (ISM) radio bands at 2.4 GHz.

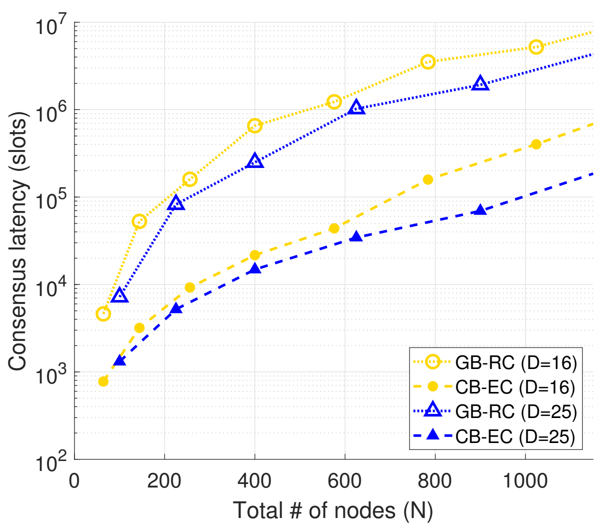
Fig. 5a illustrates the consensus latency of the conventional RC with broadcasting (B-RC) and the upper-bound of the GB-RC latency for given numbers of nodes D in each local network versus total number of nodes N. In particular, although the broadcasting based RC is efficient small-scale network (small N), there is a scalability issue due to the limitation of message dissemination in a large-scale network (large N), as shown in the figure. Find the GB-RC latency is much lower than that of B-RC in most cases, since the GB-RC leverages multi-hop to disseminate messages to overcome the broadcast distance limitation. This means that the GB dissemination method well combines two methods for large-scale networks.

Fig. 5b illustrates the consensus latency of the GB-RC and latency upper-bound of the CB-EC versus the total number of nodes N for given numbers of nodes D in each local network. From the figure, we can easily see that the GB-RC latency is much higher compared to the latency of the CB-EC. The required number of time slots in the GB-RC is about 55x larger than that of CB-EC when $N = 600$. Furthermore, the slope of the latency with respect to the total number of nodes is not largely affected by D in the GB-RC, however, in the CB-EC, the slope becomes smaller when D increases. This reveals that the efficiency of the CB dissemination in the CB-EC is more pronounced if D increases and the dissemination latency inside a local network increases. Moreover, we can observe that the latency of GB-RC can also be reduced by increasing D since the TDMA parameter T and number of relay nodes are reduced. Moreover, we can observe that the consensus latency is high for low N region when D is large. This comes from the efficiency of the CB dissemination in the CB-EC which is more pronounced if D increases. However, As the node number increases while D is fixed, the latency of CB-EC will also increase rapidly when network becomes even larger. This is straight forward since diversity gain is limited while D is finite. Therefore applying the hierarchical structure with the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3118928, IEEE Internet of Things Journal



(a)



(b)

Fig. 5: The consensus latency of (a) the broadcasting based RC (B-RC), GB-RC and (b) latency of the GB-RC, CB-EC versus the total number of nodes (N) with given number of nodes (D) in the local network.

CB-EC can be interesting direction to investigate consensus protocol for even larger network.

Fig.6 illustrates the consensus latency of the conventional RC with broadcasting (B-RC), GB-RC and CB-EC latency versus node density for given network size. Nodes are distributed randomly in global network of 500x500 square-meters, which is divided into $K = 100$ local networks. We plot the average value of the latency for each method by running 100 iterations of simulation. In particular, when the network size is large, B-RC mechanism shows worst performance since the communication between distant nodes are not reliable. The GB-RC latency is much lower than the B-RC latency, and since the number of nodes increases with node density, the latency of both techniques also increases naturally. The most interesting, but intuitive result is the latency of CB-EC.

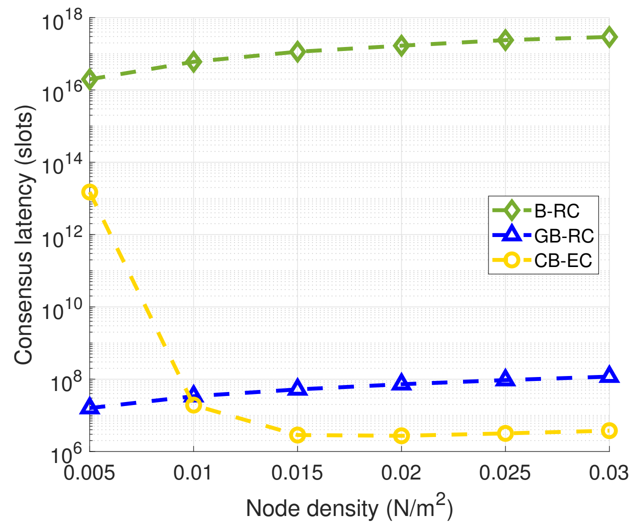


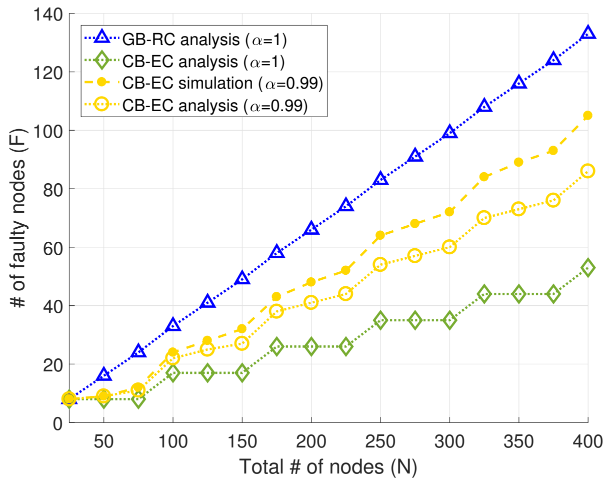
Fig. 6: The consensus latency of the broadcasting based RC (B-RC), GB-RC, and CB-EC versus node density with randomly distributed nodes in fixed network size (500x500 square-meters).

When the node density is below 0.015 per square-meter, it can be seen that the CB-EC latency decreases despite the increase of the node density. This shows that cooperative broadcast requires enough number of transmit nodes to enjoy the benefit of diversity gain. With low user density, a single local network lacks a number of nodes that can utilize cooperative-broadcasting to the entire network. However if the user number is enough to achieve diversity gain, CB-EC can reduce consensus latency even more than GB-RC.

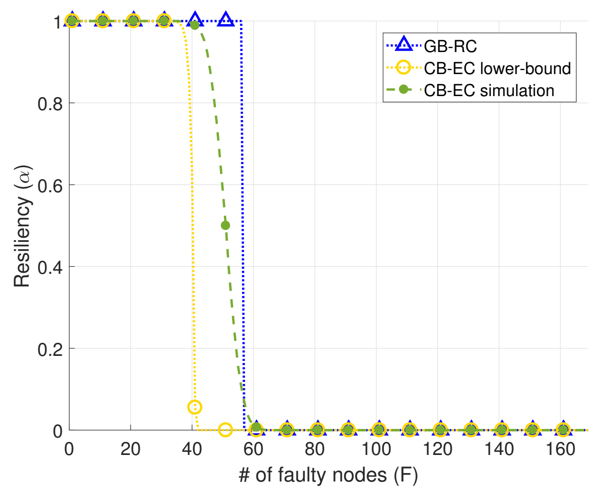
Fig. 7a illustrates the number of faulty nodes against which the GB-RC and the CB-EC are resilient versus the total number of nodes for given $D = 25$. We evaluate the maximum number of faulty nodes against which the system is resilient for the perfect resiliency case, i.e., $\alpha = 1$, in each mechanism and for a $\alpha = 0.99$ resiliency case in the CB-EC through the lower-bound obtained by analysis in the previous section and the numerical simulation by running 100000 iterations for random faulty node distribution. For the perfect resiliency case, the maximum tolerable number of faulty nodes is about 3 times higher in the GB-RC compare to the CB-EC. However, the maximum tolerable number of faulty nodes in the CB-EC increase significantly if we compromise stochastic resiliency. This means that the CB-EC can guarantee its resiliency with a high probability even though the number of faulty nodes are more than F_*. Fig. 7b illustrates the probability of resiliency of both the mechanisms for the given total number of nodes $N = 169$ and number of the local networks $K = 13$. Including the GB-RC resiliency, the lower bound of the CB-EC resiliency α and numerically simulated the CB-EC resiliency by iterating for node distribution 10000 times are provided in the figure. It shows that the GB-RC is perfectly resilient while faulty node number satisfies $N > 3F$. On the other hand, the number of faulty node to be perfectly resilient in CB-EC is lower than that of GB-RC. However, it can be seen that the maximum number of faulty nodes satisfying the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3118928, IEEE Internet of Things Journal

high resiliency probability, $\alpha > 0.99$, is almost $F = 40$ which is significantly larger than $F_* = 25$, the maximum number of faulty node for the perfect resiliency $\alpha = 1$. In addition, it can be confirmed that the theoretically analyzed lower bound is tight to the simulation result in the range where the α is large. This means that the lower-bound of α provides a meaningful number of tolerable faulty nodes in the high resiliency-required applications.



(a)



(b)

Fig. 7: (a) Maximum tolerable number of faulty nodes versus the total number of nodes N for $\alpha = 1, 0.99$-resiliency in the GB-RC and CB-EC. (b) Resiliency probability α versus the number of faulty nodes for given total number of nodes $N = 169$ and local networks $K = 13$.

VI. Discussion & Conclusion

In this paper, we proposed and investigated a couple of distributed consensus mechanisms for large-scale wireless permissioned networks. We extended the conventional RC and made it suitable to large-scale networks by using the GB message dissemination method, and thereby proposed the GB-RC. Furthermore, to overcome the fundamental limits, such as latency and scalability issues of the conventional RC and GB-RC mechanisms, we proposed the CB-EC utilizing the CB message dissemination method that exhibits lower consensus latency in a large-scale network by compromising resiliency against the faulty nodes. The consensus latency and resiliency of each method were mathematically analyzed and evaluated through numerical simulations. Consequently, we claim that under moderate environments, where there exists a limited number of faulty nodes in the network, the CB-EC is the fastest and scalable among the other existing methods in the large-scale wireless networks.

Besides, the extension of the proposed mechanisms can be done towards more practical scenarios. To advocate such a potential, we conclude this article by listing promising directions for future research.

- The structure of CB-EC can be extended to more than 2 levels of hierarchical structure network, i.e., extending the current 2 level hierarchy model of local and global networks. This will result in more efficient distributed consensus and communication in larger networks.
- The introduced dissemination methods, i.e., GB and CB, can be jointly designed to reduce consensus latency according to the node density in the network.
- Leveraging other multiple access methods, such as FDMA, CDMA and OFDMA, during message dissemination and various network division rule may be able to reduce consensus latency in specific scenarios.
- Taking more sophisticated network topology into account for analyzing wireless distributed consensus mechanism can be another challenge.
- Last, the proposed wireless distributed consensus mechanism can be applied to the existing research of Blockchain technologies for IoT devices which is investigated under wired condition [29], [30], thereby obtaining well-suited Blockchain technologies for wireless IoT networks.

Appendix A
Proof of Theorem 1

Consider a message delivery from an arbitrary source node i to the destination node j, where $i, j \in \mathcal{N}, i \neq j$. A route of the message delivery from node i to j can be seen as a path between the two nodes. A single message may flow over various paths and each path will have different latency. Since we use the GB dissemination in the GB-RC, the latency of each path depends on the number of passing local networks in each path. Among these various paths, the path which passes the least number of local networks will be the shortest path to deliver message from i to j . Let $\mathcal{K}_{ij}$ denote the set of local networks which the shortest path from i to j passes and let $\mathcal{R}_k$ denote the relay node set of the local network $k, \forall k \in \mathcal{K}_{ij}$. Then we can represent the set of local networks in the shortest path from node i to a most distant node as $\mathcal{K}_{i\max} = \underset{\mathcal{K}_{ij}, j\in\mathcal{N}}{\arg\max} |\mathcal{K}_{ij}|$. Using these notations, we can obtain the required number of time slots for the message dissemination of node i, τ_i, as

$$\tau_i = x_i + \sum_{k\in\mathcal{K}_{i\max}\setminus\hat{k}_i} \sum_{r\in\mathcal{R}_k} x_r + x_{\hat{r}_i} \tag{A.1}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3118928, IEEE Internet of Things Journal

where $\hat{k}_i$ denote the most distant local network among $\mathcal{K}_{i,\max}$ and $\hat{r}_i$ is the last relay node that broadcasts the message to the most distant node from i. As mentioned in Sec. III-C, each local network is virtually divided into square-shaped region with equal size. During the GB dissemination, a transmit node has to broadcast message to all nodes with its region and 8 neighboring regions. Although transmit nodes at the edge regions need to disseminate the message to less regions, all the transmit nodes are assumed to disseminate the message to nodes in adjacent 9 regions, which results in an upper bound on the latency. For tractable analysis, we further consider the worst case that the destination node is placed at the maximum distance is $R(\sqrt{2D}-\sqrt{2})$ for each broadcast which can be written as

$$\epsilon_{\max} = 1-\exp\left(-10^{\frac{\mathrm{PL}_{dB}(R_0)}{10}}\rho\frac{P_n}{P_t}\left(\frac{R(\sqrt{2D}-\sqrt{2})}{R_0}\right)^{\eta}\right). \tag{A.2}$$

Finally, using the required number of time slots for individual-broadcasting in (10), we can derive the upper bound of the GB-RC latency as

$$\begin{aligned} L_{\text{GB-RC}} &= T\tau_o\sum_{i\in\mathcal{N}}\tau_i && \text{(A.3)}\\ &= T\tau_0\sum_{i\in\mathcal{N}}\left(x_i+\sum_{k\in\mathcal{K}_{i\max}\setminus\hat{k}_i}\sum_{r\in\mathcal{R}_k}x_r+x_{\hat{r}_i}\right) && \text{(A.4)}\\ &\le T\tau_o\sum_{i\in\mathcal{N}}\left(g\left(\frac{9D}{4}-1,\epsilon_{i\max}\right)\right.\\ &\quad+\sum_{k\in\mathcal{K}_{i\max}\setminus\hat{k}_i}\sum_{r\in\mathcal{R}_k}g\left(\frac{9D}{4}-1,\epsilon_{r\max}\right) && \text{(A.5)}\\ &\quad\left.+g\left(\frac{9D}{4}-1,\epsilon_{\hat{r}_i\max}\right)\right)\\ &\le T\tau_o\sum_{i\in\mathcal{N}}\left(4|\mathcal{K}_{i\max}|-2\right)g\left(\frac{9D}{4}-1,\epsilon_{\max}\right) && \text{(A.6)}\\ &= TN\left(4|\bar{\mathcal{K}}_{\max}|-2\right)g\left(\frac{9D}{4}-1,\epsilon_{\max}\right)\tau_o && \text{(A.7)} \end{aligned}$$

where $T\tau_o$ is due to the T-TDMA scheme and the single time slot length. Since $|\mathcal{K}_{i\max}|$ depends on the local network number K, we can rewrite the value $\sum_{i\in\mathcal{N}}|\mathcal{K}_{i\max}|$ using the mean value for all $i\in\{1,...,N\}$, i.e $N|\bar{\mathcal{K}}_{\max}|$, for given K. The mean value $|\bar{\mathcal{K}}_{\max}|$ is derived in the square-shaped network as follow.

$$|\bar{\mathcal{K}}_{\max}| = \begin{cases} \frac{5\sqrt{K}}{6}-\frac{1}{3\sqrt{K}}+\frac{1}{2}, & \sqrt{K}\text{ is even}\\ \frac{5\sqrt{K}}{6}-\frac{11}{6\sqrt{K}}+\frac{3}{2K}+\frac{1}{2}, & \sqrt{K}\text{ is odd}\end{cases} \tag{A.8}$$

Appendix B
Proof of Theorem 2

The consensus latency of the CB-EC can be analyzed separately for each phase. As mentioned earlier, we neglect the local computation time during the consensus process. Thus, the following three factors will be incorporated in the latency of the CB-EC.

Proposal latency (τ_p): The proposal latency τ_p for arbitrary node i in local network k can be obtained as

$$\tau_p = T\tau_o\left(g(D-1,\epsilon_{i\max})+2Dg(N-D,\bar{\epsilon}_{(k)\max})\right) \tag{B.1}$$

where the first term in the brackets denotes the required number of time slots for node i to broadcast the message to local network k via individual-broadcasting (10), and the second term denotes the required number of time slots for global network dissemination from local network k via the CB dissemination derived from (21). Moreover, $T\tau_0$ comes from the T-TDMA scheme and the single time slot length. The proposal latency depends on the node location and the worst case of the proposal latency is when i is located on the edge of the square-shaped global network. Therefore we can upper bound the proposal latency τ_p as follow.

$$\tau_p \le T\tau_o\left(g(D-1,\epsilon_{\max})+2Dg(N-D,\bar{\epsilon}_{\max})\right) \tag{B.2}$$

where $\epsilon_{\max}$ is (A.2) since the maximum distance inside a local network is $R(\sqrt{D}-\sqrt{2})$. $\bar{\epsilon}_{\max}$ can be obtained as (50) by applying the maximum distance between an arbitrary local network and an arbitrary node,$\sqrt{2}R(2\sqrt{N}-\sqrt{D}-1)/2$, to (33).

Local commit latency (τ_l): Each node in a local network, except the proposer i, will commit its local validation to the other nodes in the same local network. Since this phase is executed in parallel on each local network, the upper bound of τ_l can be obtained from the upper bound of local commit latency in an arbitrary local network q.

$$\begin{aligned} \tau_l &= T\tau_o\sum_{v\in\mathcal{N}_q}g(D-1,\epsilon_{v\max}) && \text{(B.3)}\\ &\le T\tau_o Dg(D-1,\epsilon_{\max}) && \text{(B.4)} \end{aligned}$$

where (B.3) is the summation of required time duration for every node's individual-broadcasting derived from (10) when the target nodes are the other nodes in transmitter node's local network. (B.4) is the upper bounded latency by assuming every node's outage probability is $\epsilon_{\max}$.

Global commit latency (τ_g): During the global commit phase, each local network will transmit its sub-global validation through CB. The CB dissemination is carried out for each local network in a predetermined order. Thus we can derive the upper bound of τ_g as

$$\begin{aligned} \tau_g &= T\tau_o\sum_{q\in\mathcal{K}}2Dg(N-D,\bar{\epsilon}_{(q),\max}) && \text{(B.5)}\\ &\le T\tau_o\left(2DKg(N-D,\bar{\epsilon}_{\max})\right) && \text{(B.6)} \end{aligned}$$

where (B.5) is the summation of required number of time slots for every local network $q\in\mathcal{K}$ via CB to the other $N-D$ nodes derived as (21) and (B.6) is the upper bounded latency by assuming every CB dissemination outage probability is $\bar{\epsilon}_{\max}$.

Consequently, the upper bound of the CB-EC latency becomes

$$\begin{aligned} L_{\text{CB-EC}} &= \tau_p+\tau_l+\tau_g && \text{(B.7)}\\ &\le T((D+1)g(D-1,\epsilon_{\max})\\ &\quad+2D(K+1)g(N-D,\bar{\epsilon}_{\max}))\tau_o. && \text{(B.8)} \end{aligned}$$

References

[1] D. Evans, "How the next evolution of the internet is changing everything," 2011.

[2] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5g be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, 2014.

[3] H. Seo, J. Hong, and W. Choi, "Low latency random access for sporadic mtc devices in internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5108–5118, 2019.

[4] Y. Kang, H. Seo, and W. Choi, "When to realign the receive beam in high mobility v2x communications?" *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 13 180–13 195, 2020.

[5] Infosys, "Smart connected stadiums, smart venues, revolutionary experiences," *White paper, 2018. [Online]. Available: https://www.infosys.com/engineering-services/white-papers/documents/smart-connected-stadiums.pdf.*

[6] X. Krasniqi and E. Hajrizi, "Use of iot technology to drive the automotive industry from connected to full autonomous vehicles," *IFAC-PapersOnLine*, vol. 49, no. 29, pp. 269–274, 2016, 17th IFAC Conference on International Stability, Technology and Culture TECIS 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896316325162

[7] Z. Yuan, J. Jin, L. Sun, K.-W. Chin, and G.-M. Muntean, "Ultra-reliable iot communications with uavs: A swarm use case," *IEEE Communications Magazine*, vol. 56, no. 12, pp. 90–96, 2018.

[8] G. Sushanth and S. Sujatha, "Iot based smart agriculture system," in *2018 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, 2018, pp. 1–4.

[9] S. Aheleroff, X. Xu, Y. Lu, M. Aristizabal, J. Pablo Velásquez, B. Joa, and Y. Valencia, "Iot-enabled smart appliances under industry 4.0: A case study," *Advanced Engineering Informatics*, vol. 43, p. 101043, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474034620300124

[10] H. Seo, J. Park, M. Bennis, and W. Choi, "Communication and consensus co-design for distributed, low-latency, and reliable wireless systems," *IEEE Internet of Things Journal*, vol. 8, no. 1, pp. 129–143, 2021.

[11] M. Castro and B. Liskov, "Practical byzantine fault tolerance," *OSDI*, vol. 99, pp. 173–186, 1999.

[12] M. Doddavenkatappa, M. C. Chan, and B. Leong, ""splash: Fast data dissemination with constructive interference in wireless sensor networks," in *USENIX NSDI*, 2013.

[13] B. Sirkeci-Mergen, A. Scaglione, and G. Mergen, "Asymptotic analysis of multistage cooperative broadcast in wireless networks," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2531–2550, 2006.

[14] A. Özgür, O. Lévêque, and D.Tse, "Hierarchical cooperation achieves optimal capacity scaling in ad hoc networks," *IEEE Transactions on Information Theory*, vol. 53, no. 10, pp. 3549–3572, 2007.

[15] R. Eletreby, D. Zhang, S. Kumar, and O. Yağan, "Empowering low-power wide area networks in urban settings," in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM '17, 2017, p. 309–321.

[16] R. Kotla, L. Alvisi, M. Dahlin, A. Clement, and E. Wong, "Zyzzyva: speculative byzantine fault tolerance," *ACM Transactions on Computer Systems*, vol. 27, no. 7, pp. 1–39, 2010.

[17] R. Guerraoui, N. Knežević, V. Quéma, and M. Vukolić, "The next 700 bft protocols," *ACM Proceedings of the 5th European Conference on Computer Systems*, pp. 363–376, 2010.

[18] L. Lamport, R. Shostak, and M. Pease, "The byzantine generals problem," *ACM Trans. Program. Lang. Syst.*, vol. 4, no. 3, p. 382–401, 1982.

[19] L. Lamport, "The part-time parliament," *ACM Transactions on Computer Systems*, vol. 16, no. 2, pp. 133–169, 1998.

[20] L. Lamport, "Paxos made simple," *SIGACT*, vol. 32, no. 4, pp. 18–25, 2001.

[21] M. J. Fischer, N. A. Lynch, and M. S. Paterson, "Impossibility of distributed consensus with one faulty process," *J. ACM*, vol. 32, no. 2, pp. 374–382, Apr. 1985. [Online]. Available: http://doi.acm.org/10.1145/3149.214121

[22] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *http://bitcoin.org/bitcon.pdf.*

[23] L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A secure sharding protocol for open blockchains," *ACM SIGSAC Conference on Computer and Communications Security*, pp. 17–30, 2016.

[24] E. Kokoris-Kogias, P. Jovanovic, L. Gasser, N. Gailly, E. Syta, and B. Ford, "Omniledger: A secure, scale-out, decentralized ledger via sharding," in *2018 IEEE Symposium on Security and Privacy (SP)*, 2018, pp. 583–598.

[25] H. Dang, T. T. A. Dinh, D. Loghin, E.-C. Chang, Q. Lin, and B. C. Ooi, "Towards scaling blockchain systems via sharding," in *Proceedings of the 2019 International Conference on Management of Data*, ser. SIGMOD '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 123–140.

[26] M. J. Amiri, D. Agrawal, and A. E. Abbadi, "Sharper: Sharding permissioned blockchains over network clusters," 2020.

[27] Z. Jiang, Z. Cao, B. Krishnamachari, S. Zhou, and Z. Niu, "Senate: A permissionless byzantine consensus protocol in wireless networks for real-time internet-of-things applications," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6576–6588, 2020.

[28] B. Cao, Y. Li, L. Zhang, L. Zhang, S. Mumtaz, Z. Zhou, and M. Peng, "When internet of things meets blockchain: Challenges in distributed consensus," *IEEE Network*, vol. 33, no. 6, pp. 133–139, 2019.

[29] B. Cao, Z. Zhang, D. Feng, S. Zhang, L. Zhang, M. Peng, and Y. Li, "Performance analysis and comparison of pow, pos and dag based blockchains," *Digital Communications and Networks*, vol. 6, no. 4, pp. 480–485, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352864819301476

[30] Y. Li, B. Cao, M. Peng, L. Zhang, L. Zhang, D. Feng, and J. Yu, "Direct acyclic graph-based ledger for internet of things: Performance and security analysis," *IEEE/ACM Transactions on Networking*, vol. 28, no. 4, pp. 1643–1656, 2020.

[31] H. Seo, J. Park, M. Bennis, and W. Choi, "Consensus-before-talk: Distributed dynamic spectrum access via distributed spectrum ledger technology," in *2018 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2018, pp. 1–7.

[32] P. Danzi, A. E. Kalør, Č. Stefanović, and P. Popovski, "Delay and communication tradeoffs for blockchain systems with lightweight iot clients," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2354–2365, 2019.

[33] Y. Liu, K. Wang, Y. Lin, and W. Xu, "LightChain: A lightweight blockchain system for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3571–3581, 2019.

[34] H. Kim, J. Park, M. Bennis, and S. Kim, "Blockchained on-device federated learning," *IEEE Communications Letters*, vol. 24, no. 6, pp. 1279–1283, 2020.

[35] G. Lee, J. Park, W. Saad, and M. Bennis, "Performance analysis of blockchain systems with wireless mobile miners," *IEEE Networking Letters*, vol. 2, no. 3, pp. 111–115, 2020.

[36] F. B. Schneider, "Implementing fault-tolerant services using the state machine approach: A tutorial," *ACM Computing Surveys (CSUR)*, vol. 22, no. 4, pp. 299–319, 1990.

[37] G. Miao, J. Zander, K. W. Sung, and S. Ben Slimane, *Fundamentals of Mobile Data Networks*. Cambridge University Press, 2016.

[38] A. Mahmood, M. I. Ashraf, M. Gidlund, and J. Torsner, "Over-the-air time synchronization for urllc: Requirements, challenges and possible enablers," in *2018 15th International Symposium on Wireless Communication Systems (ISWCS)*, Aug 2018, pp. 1–6.

[39] B. R. Calder and A. McLeod, "Ultraprecise absolute time synchronization for distributed acquisition systems," *IEEE Journal of Oceanic Engineering*, vol. 32, no. 4, pp. 772–785, Oct. 2007.

[40] G. Tsudik, "Message authentication with one-way hash functions," *ACM SIGCOMM Computer Communication Review*, vol. 22, no. 5, pp. 29–38, 1992.

[41] N. community, "Nxt: a peer-to-peer digital socioeconomic system," *White paper, July. 2014*.

[42] S. Popov, "The tangle," *White paper, 2018. [Online]. Available: https://www.iota.org/research/academic-papers.*

[43] Ethereum Community, "A next-generation smart contract and decentralized application platform," *White Paper, available at https://github.com/ethereum/wiki/wiki/White-Paper*.

[44] J. Grönkvist, A. Komulainen, U. Sterner, and U. Uppman, "Dynamic scheduling for cooperative broadcasting in tactical ad hoc networks," in *MILCOM 2016 - 2016 IEEE Military Communications Conference*, 2016, pp. 1034–1040.

[45] X.-B. Liang, "Orthogonal designs with maximal rates," *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2468–2503, 2003.

[46] A. Goldsmith, *Wireless Communications*, 1st ed. Cambridge University Press, Aug. 2005.