Performance Analysis of Wireless Practical Byzantine Fault Tolerance Networks Using IEEE 802.11

Ziyi Zhou, Oluwakayode Onireti, Lei Zhang and Muhammad Ali Imran
James Watt School of Engineering, University of Glasgow, G12 8QQ, Glasgow, UK
Email: 2289312z@student.gla.ac.uk, {Oluwakayode.Onireti, Lei.Zhang and Muhammad.Imran}@glasgow.ac.uk

2021 IEEE Globecom Workshops (GC Wkshps) | 978-1-6654-2390-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/GCWkshps52748.2021.9682068

***Abstract*—Blockchain has achieved great success in cryptocurrency for its peculiarities for security and privacy, which are also important in the wireless network. Therefore, there are growing interests in applying blockchain to the wireless network. Wireless Practical Byzantine Fault Tolerance (PBFT) is considered the most applicable consensus mechanism. However, the existing researches and applications are mostly under wired scenarios. In this paper, we investigated the performance of the wireless PBFT network using IEEE 802.11 under unsaturated situations. The performance is evaluated through three metrics: success probability, delay and throughput. Results suggest that there exists a minimum transmission success probability to achieve the end-to-end performance required for the PBFT consensus protocol.**

***Index Terms*—Blockchain, PBFT, IEEE 802.11, Internet of things**

I. Introduction

The past few decades have witnessed tremendous growth in mobile users, which has also driven the rapid development of wireless networks. According to a report from IBM, there are currently over 25 billion connected devices in 2020 and this is forecasted to surpass 100 billion by 2050 [1]. This will indeed pose a considerable challenge for the wireless network. At present, wireless networks operate a centralized structure, where the user's data are stored in large-scale cloud centers. Such a centralized network framework faces ever-increasing severe problems, such as hacking, high maintenance fees, low scalability, and low recovery ability.

The advent of blockchain has paved a possible way for future wireless networks. As an emerging technique, blockchain is a distributed ledger that allows peer-to-peer communication in a trustless network without the involvement of a third party. Due to its merits of decentralization, security, robustness, and resilience, it has been deemed as a favourable solution for the next-generation wireless network. The smart contracts enabled consensus mechanism (CM) is the basis of blockchain, and it ensures its effectiveness. Moreover, Practical Byzantine Fault Tolerance (PBFT) [2], a voting-based CM, is well suited for wireless networks due to its high transaction throughput, low computational requirement, and low complexity. PBFT provides safety and liveness, with a fault tolerance of $\frac{n-1}{3}$, where n is the total number of nodes. Such traits make it appealing to future wireless networks. However, compared with the most famous application of blockchain (Bitcoin), the research of blockchain in the wireless network remains unexplored since the existing works mainly focus on the wired scenario. The connections among nodes under a wireless environment are deemed unstable due to the limited spectrum resources, varying channel strength, and various channel topologies. Therefore, it is important to investigate the impact of wireless communications on the performance of PBFT-based blockchain networks.

In [3], the authors Onireti *et al.* investigated the viable area of the PBFT wireless network. They proposed the new concept of the viable area that guarantees the minimum number of replica nodes required for achieving the safety and liveness of the PBFT wireless network. In [4], Zhang *et al.* discussed the resources required to run the wireless blockchain network (WBN). They proposed a novel standard to measure the performance of a WBN in terms of four metrics: communication complexity and spectrum requirement, communication reliability /receiver sensitivity, number of nodes/replicas, and transmission power. The work in [4] has provided a new perspective of WBN performance measurement. In [5], the authors proposed a scalable multi-layer PBFT based consensus mechanism, which greatly reduces communication complexity. The authors in [6] investigated how carrier sense multiple access/collision avoidance (CSMA/CA) affects the performance and security in wireless blockchain networks.

In this paper, we investigate the performance of a wireless PBFT which is implemented over the well known IEEE 802.11 protocol [7]. The IEEE 802.11 is a set of communication standards that have been widely used by wireless local area networks (WLAN). It embraces a distributed coordination function (DCF) that is similar to carrier sense multiple access with collision avoidance (CSMA/CA) techniques. In this work, we utilize the IEEE 802.11 broadcast scheme in all the phases of the wireless PBFT; hence, no acknowledgment (ACK) is transmitted by any of the nodes in the network. Further, there are no exchange of request to send (RTS)/clear to send (CTS) messages. However, physical sensing is still applied in our model. Our network environment is derived from [7], which considered unsaturated traffic as in real wireless networks. Based on this assumption, we evaluate the performance of the wireless PBFT using the IEEE 802.11 protocol through three metrics: success probability, average transaction confirmation delay, and average transaction throughput. More specifically,

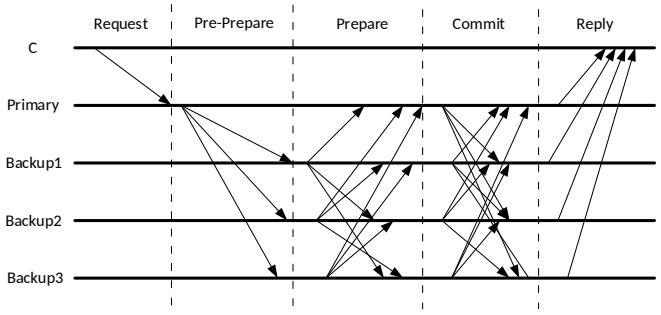

Fig. 1. The normal case operation of the PBFT network [2]

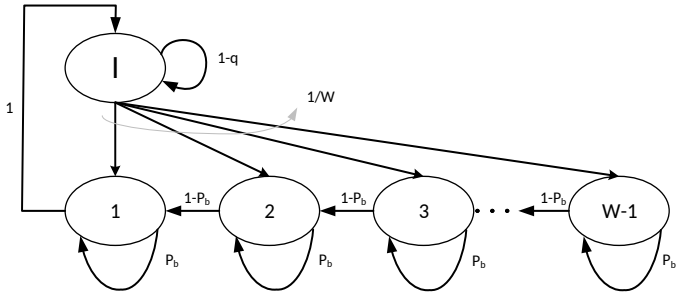

Fig. 2. Markov chain for the contention model in unsaturated traffic scenario [8]

each of them will be derived and simulated over different phases of the normal operation of PBFT: pre-prepare, prepare and commit as well as the end-to-end performance measure. According to our analysis work, these metrics are mutually decided by the message/packets arrival rate λ, contention window size W, and the number of nodes n involved in the PBFT consensus network. Our work thus presents the operational flows of PBFT over IEEE 802.11 wireless communication protocol and the factors that can affect its performance.

The paper is outlined as follows. In Section II, we present the system model and introduce the fundamentals of the PBFT consensus protocol. In addition, we also present the Markovian model of the IEEE 802.11 unsaturated traffic upon which our analysis is based on. Section III presents the analysis and mathematical derivation of the success probabilities, average transaction throughput, and average transaction confirmation delay for the wireless PBFT consensus networks using IEEE 802.11. We demonstrate numerical results in Section IV. Finally, Section V concludes the paper.

II. System Model

In this section, we discuss the fundamental operation of the PBFT consensus protocol and IEEE 802.11 protocol.

A. Practical Byzantine Fault Tolerance (PBFT)

Suppose wireless PBFT network is composed of n nodes, for a successful vote, there should not be more than f faulty or byzantine nodes, where the relationship between f and n is as follows:

$$f \leq \left\lfloor \frac{n-1}{3} \right\rfloor \tag{1}$$

As long as f is no more than $\lfloor \frac{n-1}{3} \rfloor$, the safety and liveness of PBFT will be guaranteed. In wireless PBFT, the network moves through a succession of configurations called views as it progresses, and for each view, it selects one of the nodes as the primary node while other nodes serve as backups. A complete validation process can be divided into four phases in the normal case operation of the PBFT network: pre-prepare, prepare, commit and reply (as shown in Fig. 1). Every node in the wireless PBFT network is involved in the validation process. We denote the set of nodes by $\mathcal{R}$ and identify each node by an integer in $\{0, ..., |\mathcal{R}|-1\}$. Note that even though $\mathcal{R}$ could be more than $3f+1$, additional nodes will not contribute to the performance of the PBFT network. Thus, we consider $3f+1$ as the maximum number of nodes in the network. Every node in the PBFT wireless network takes turn to be selected as a primary node, according to the view change rule. Then the primary node is denoted as v_p and can be obtained by:

$$v_p = v(\text{mod } |R|) \tag{2}$$

where v is the view number. Every time the primary node fails, the view change rule will be carried out and a new primary will be selected. After the primary node receives the request from a client, the normal operation case of the PBFT starts as illustrated in Fig. 1:

- *Pre-prepare:* The primary node broadcasts the pre-prepare message to all backups and the request message is not included to keep the message short.
- *Prepare:* The replica that receives the pre-prepare message broadcasts the prepare message, the digest of the message $D(m)$, to other replicas. If a replica has received $2f$ prepare messages that match the pre-prepare message, the prepare message can be regarded as valid.
- *Commit:* If the prepare message is true, the replica broadcasts the commit message to the rest replicas.
- *Reply:* The reply message received by the client shows the result of the request.

The result of the request is regarded as valid only if the client receives at least $f+1$ replies with the same response from the replicas.

B. Markovian Model of IEEE 802.11 under Unsaturated Traffic Scenario

In [8], Markov chain (as shown in Fig. 2) was used to model the non-saturated IEEE 802.11 broadcast scenario where n nodes simultaneously contend for the channel to transmit without hidden nodes and capture effects. Thus, all packet losses are caused by collisions. Our analysis of the wireless PBFT using the IEEE 802.11 protocol is based on this scenario. Here, a random process $b(t)$ of a node is denoted as a backoff counter, which will be decremented if an idle channel is sensed and stops when a transmission is detected. When the backoff counter reduces to zero, the node starts to transmit. The value of $b(t)$ is only related to contention window size. In wireless PBFT networks, transmissions in the pre-prepare,

prepare and commit phases are broadcast, and retransmission is not considered. Hence, the window size is always equal to the initial minimum size, which is denoted by W.

So, here we can have two assertions:

- The probability, τ that a node will attempt to transmit in a random time slot is constant for all time slots.
- The probability P_b that a collision happens in a time slot is constant and independent of the number of previous collisions.

Note that in Fig. 2 the unsaturated condition is achieved through state labelled I which accounts for the following:

- The buffer of a transmitting node is empty right after a successful transmission.
- A node is at an idle state with no packets in the buffer till when a new packet arrives for transmission.

C. Performance of IEEE 802.11

As aforementioned, a node starts transmission when the counter reaches zero. So the probability τ that a node start transmission in a randomly chosen time slot can be obtained from [7], [8] as

$$\tau = \left(\frac{1}{q} + 1 + \frac{(W-1)}{2(1-P_b)} \right)^{-1} \tag{3}$$

where P_b is the probability that the channel is busy. Given that there are n nodes in the network P_b can be expressed as

$$P_b = 1 - (1-\tau)^{n-1} \tag{4}$$

Further, the parameter q in (3) is the probability that there is at least one packet in the buffer waiting for transmission, and it can be expressed as follow:

$$q = 1 - e^{-\lambda E[S_{ts}]} \tag{5}$$

where λ represents the rate at which packets arrive at a node's buffer and $E[S_{ts}]$ is the expected time per slot, which is related to the network parameters.

Let P_t denote the probability that there is at least one node transmitting within the same slot time, where n nodes are contending for the channel. Thus, we can derive the relation between P_t and τ as follow:

$$P_t = 1 - (1-\tau)^n \tag{6}$$

Furthermore, a successful transmission occurs only if there is only one node transmitting in a time slot. Thus, the transmission success probability P_s can be expressed as:

$$P_s = \frac{n\tau(1-\tau)^{n-1}}{P_t} \tag{7}$$

The expected time per slot $E[S_{ts}]$ in (5) can be represented as:

$$E[S_{ts}] = (1-P_t)\sigma + P_t(1-Ps)T_c + P_tP_sT_s \tag{8}$$

where σ is the idle slot time, T_s is the average time that the channel is sensed busy because of successful transmission. T_c is the average time that the channel is sensed busy by each node during a collision. Note that we have the same cost for the successful and unsuccessful transmission since the broadcast does not employ the RTS/CTS mechanism or acknowledgment (ACK) [9]. Hence

$$T = T_s = T_c = \frac{H + E[P]}{R} + DIFS + \delta \tag{9}$$

where δ is the propagation delay, DIFS is the period for a distributed interframe space, and R is the system transmission rate. Note that H is the header length, which is the sum of MAC and PHY headers, and $E[P]$ refers to average packet length. Consequently, by substituting for P_s and P_t from (6) and (7), respectively, into (8), we can express the expected time slot as

$$E[S_{ts}] = (1-\tau)^n\sigma + (1-(1-\tau)^n)T. \tag{10}$$

Another important performance metric for IEEE 802.11 protocol is the medium access delay. In our framework, medium access delay refers to the period between when a node starts contending for transmission and when the packet is successfully transmitted [10]. Let D denote the delay, which can be computed as:

$$D = T_s + D_s + D_c + T_{slot} \tag{11}$$

where,

- T_s is the time taken for a successful transmission.
- D_s is the average time the channel is in use and thus sensed busy due to the successful transmission of other nodes. Assume there are i successful transmission in a round, then T_s will be :

$$D_s = T_s(i-1) \tag{12}$$

- D_c refers to the time the channel is sensed busy due to collision, i.e., unsuccessful transmission. Let $P\{N=i\}$ denote the probability that i nodes successfully broadcast their message, for a given number n overall nodes. The delay D_c as a result of collision can thus be expressed from [10] as

$$D_c = \frac{1-(1-\tau)^i - i\tau(1-\tau)^{i-1}}{\tau(1-\tau)^{i-1}} T_c \tag{13}$$

- T_{slot} is the total number of idle time slots and it can be expressed as

$$T_{slot} = \frac{1-\tau}{\tau}\sigma \tag{14}$$

By substituting for D_s, D_c and T_{slot}, from (12), (13) and (14), respectively, into (11), we can obtain

$$D = iT_s + \frac{1-(1-\tau)^i - i\tau(1-\tau)^{i-1}}{\tau(1-\tau)^{i-1}} T_c + \frac{1-\tau}{\tau}\sigma \tag{15}$$

In the next section, we build of the performance analysis of the IEEE 802.11 and derive the success probability, transaction confirmation delay and the transaction throughput when the protocol is utilized for the wireless PBFT network.

III. PBFT Networks Over IEEE 802.11

In this section, we will analyze the performance of the wireless PBFT networks using the IEEE 802.11 protocol. In particular, we derive the success probability at the phases of the wireless PBFT. Moreover, using the success probability, we also derive the average transaction confirmation delay and the transaction throughput.

A. Wireless PBFT Networks Consensus Success Probability

Even though PBFT can tolerate up to $\lfloor \frac{n-1}{3} \rfloor$ fault nodes, some non-faulty nodes may still not be able to participate in the consensus process due to the nature of the wireless network, such as channel loss or collisions. However, only an agreed valid transaction can be added to the blockchain, so the success probability of each transaction significantly affects its effectiveness. Our analysis here accounts for failure due to collisions as a result of channel contention in the IEEE 802.11 protocol. We also discuss the success probability of each phase of the wireless PBFT specifically.

1) Success Probability of Pre-prepare: At this stage, after receiving the request from the client, the primary node broadcast this request to the rest of the replicas as shown in Fig. 1. There is no contention at this stage; thus the success probability is considered to be 100%[1].

2) Success Probability of Prepare: Given the 100% success rate at the pre-prepare phase, $n-1$ nodes receive the pre-prepare message from the primary node. Then, each replica node broadcasts the digest of the message to the other replicas. To ensure the commit phase can be successfully reached, at least $2f$ nodes must receive the message from other nodes. Let us suppose that i nodes successfully broadcast their message, $P\{N = i\}$ is the conditional probability which can be expressed as

$$P\{N=i\} = \binom{n-1}{i} P_s^i (1-P_s)^{n-1-i} \tag{16}$$

Any i that is equal to or greater than $2f$ will be regarded as valid. Thus, by summing up all the valid condition probability of i, we can derive the success probability expression of the prepare phase:

$$P_p = \begin{cases} 0 & n < 2f \\ \sum_{i=2f}^{n-1} \binom{n-1}{i} P_s^i (1-P_s)^{n-1-i} & n \geq 2f \end{cases} \tag{17}$$

3) Success Probability of Commit: Commit phase is very similar to prepare. Any node that receives prepare message will broadcast the commit message to the rest nodes. The only difference is that the primary node also needs to broadcast a message in the commit phase. Therefore, the success probability expression of the commit phase can be formulated as:

$$P_c = \begin{cases} 0 & n < 2f \\ \sum_{m=2f+1}^{n} \binom{n}{m} P_s^m (1-P_s)^{n-1-m} & n \geq 2f \end{cases} \tag{18}$$

[1]Note that for tractability, we consider that only the n nodes involved in the consensus are active on the wireless network.

TABLE I
NETWORK PARAMETERS

MAC header	24 bytes
PHY header	16 bytes
Payload size	1023 bytes
Channel Bit Rate	1 Mbits/s
σ	1 μs
Slot time	20 μs
SIFS	10 μs
DIFS	50 μs

4) End-to-End Success Probability: Successful validation of a new block requires both the prepare and commit phases to be successful. Thus, the condition that at least $2f$ nodes at prepare and $2f+1$ nodes at commit receive the message must be satisfied. So, the end-to-end success probability can be expressed as (19), shown at the top of the next page.

B. Wireless PBFT Network Average Transaction Confirmation Delay and Throughput

In wireless PBFT networks, transaction confirmation delay and transaction throughput are two important metrics. The delay presented in Section II-C is the time between two successful transmissions. Meanwhile, the transaction confirmation delay in the PBFT network refers to the time between two successful consensuses. Thus, we can have the average medium access delay for prepare and commit stage, D_P and D_C:

$$D_{avr} = \sum_{i=\mathcal{F}}^{n} P\{N=i\} (iT_s + \frac{1-(1-\tau)^i - i\tau(1-\tau)^{i-1}}{\tau(1-\tau)^{i-1}} T_c + \frac{1-\tau}{\tau}\sigma) \tag{20}$$

where $\mathcal{F} = 2f$ and $2f+1$ in the prepare and commit phases, respectively, and $P\{N=i\}$ is given in (16). The expression of end-to-end delay is given in (21), shown on the top of the next page.

According to the definition of the transaction confirmation delay, we know that delay is equal to the time needed for a consensus for a new validation. Thus, the transaction throughput can be expressed as the rate of reaching a consensus which is thus given as:

$$S = \frac{1}{D_e} \tag{22}$$

where D_e is obtained from (21).

IV. Numerical Results

This section presents numerical results to illustrate our theoretical analysis and derivations. In our model, the wireless PBFT network is in a fully connected topology such that every node can transmit and receive messages directly from other nodes. The data transmission rate R is 1 Mbps. Other system parameters are in Table I:

Fig. 3 shows the relationship between success probabilities (transmission, prepare, commit and end-to-end) and the number of nodes, with packet arrival rates $\lambda = 20$ and contention window size $W = 64$. It can be seen that as the number of nodes increases, the transmission success probability P_s gradually decreases from approximately 0.95 to 0.67. This is

$$P_e = \begin{cases} 0 & n < 2f+1 \\ \sum_{i=2f}^{n-1} \sum_{m=2f+1}^{n} \binom{n}{i} P_s^i (1-P_s)^{n-1-i} \binom{n}{m} P_s^m (1-P_s)^{n-1-m} & n \geq 2f+1 \end{cases} \tag{19}$$

$$D_e = \sum_{i=2f}^{n-1} \sum_{m=2f+1}^{n} \binom{n}{i} P_s^i (1-P_s)^{n-1-i} \binom{n}{m} P_s^m (1-P_s)^{n-1-m} \left(D_{cp} + D_{cc} + (m+i)T_c + 2\frac{1-\tau}{\tau}\sigma \right) \tag{21}$$

where D_{cp} and D_{cc} (obtained from (13)) are the time taken when the channel is sensed busy due to collision in prepare and commit phases, respectively.

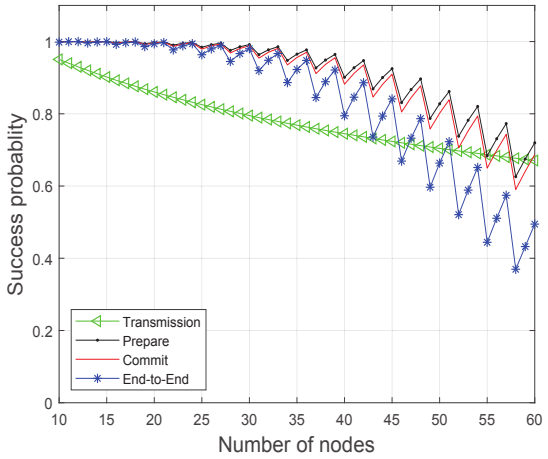

Fig. 3. Success probability of the wireless PBFT network versus number of nodes for $\lambda = 20$ and $W = 64$

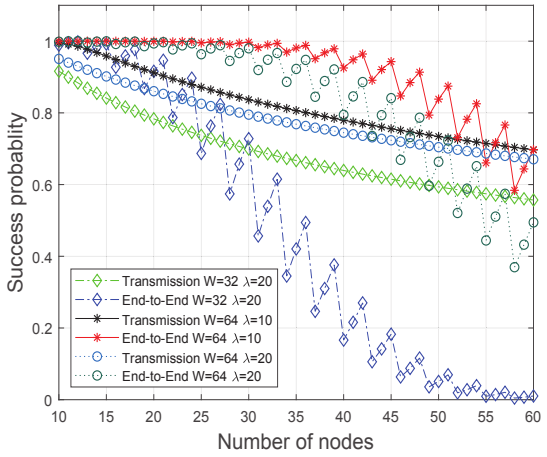

Fig. 4. Success probability comparison of the wireless PBFT network

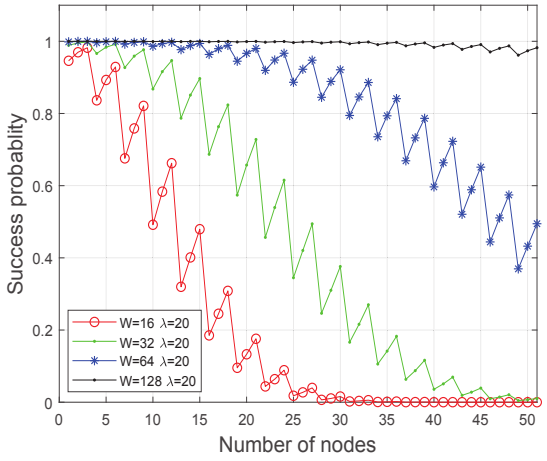

Fig. 5. End-to-end success probability for different window size W and fixed packet arrival rate $\lambda = 20$

due to the increasing likelihood of a collision as the number of nodes increases. Alongside that, after a period of levelling out, the success probability of prepare, commit, and end-to-end starts to drop when transmission success probability reaches 0.84. This indicates that the wireless PBFT network is tolerant to a certain degree of loss. Note that the plots of the success probability in the prepare and commit phases are similar because there is just one node difference between the two phases.

To explore the impact of the packet arrival rate λ and the contention window size W on the success probability, we reduce λ to 10, i.e., $[W = 64, \lambda = 10]$, and W to 32, i.e., $[W = 32, \lambda = 20]$ as shown in Fig. 4. The plot for the network with $[W = 64, \lambda = 20]$ is also shown as the benchmark. In Fig. 4, even though the transmission success probability difference from the benchmark is marginal (less than 0.1), the end-to-end transmission success probability experiences a huge difference. The lowest point on the plot of the end-to-end success probability for the network with $[W = 64, \lambda = 10]$ (red line) is around 0.6, while that of the network with $[W = 32, \lambda = 20]$ (blue line) reaches 0. Furthermore, it can be seen that, the end-to-end success probability of the wireless PBFT network is very sensitive to the transmission success probability P_s when $P_s < 0.84$, which we refer to as the critical point. So, we can have a hypothesis that W has stronger impacts on the wireless PBFT networks' performance. The results in Figs. 5 and 6 have validated this hypothesis. From Fig. 5, where $\lambda = 20$, we can see that W has a great influence on the success probability, which remains nearly 100% for $W = 128$. It can be seen that reducing W by a factor of half leads to a significant reduction in the end-to-end success probability. Especially, the case with $W = 16$ hits zero end-to-end success probability when the number of nodes only

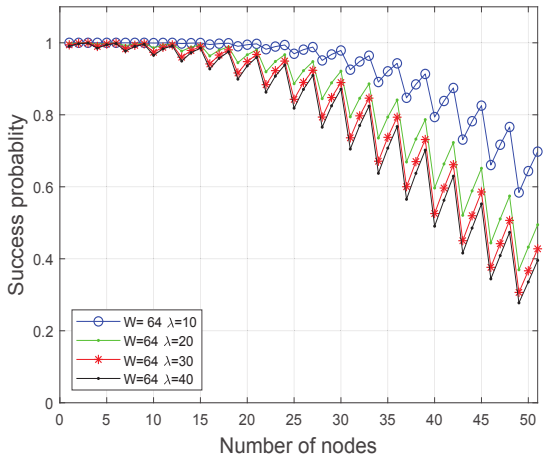

Fig. 6. End-to-end success probability for different packet arrival rate λ and fixed window size $W = 64$

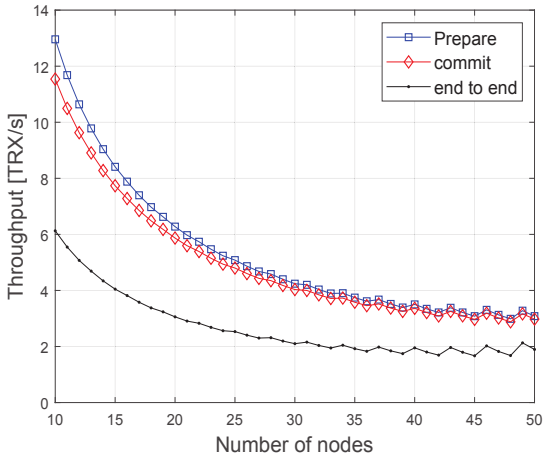

Fig. 7. Transaction throughput versus number of nodes with $\lambda = 20$ and $W = 64$

equals 25. This means the wireless PBFT network under such parameters has poor scalability. However, the difference in Fig. 6 where the contention window size is fixed to $W = 64$ can be seen to be marginal.

Figs. 7 and 8 show the relationship between the transaction throughput and transaction confirmation delay of wireless PBFT network and the numbers of nodes. Since the throughput and delay are highly related, we focus our discussion here on the throughput. In Fig. 7, after experiencing a sharp decline, it can be seen that the transaction throughput starts to converge to a fixed value when the number of nodes reaches 30. Thus, combining this with other results presented earlier above, we can maximize the performance when designing wireless PBFT network using IEEE 802.11 protocol.

V. Conclusion

The paper investigates the performance of the wireless PBFT network over wireless protocol IEEE 802.11. The analysis and simulation results have shown that PBFT can achieve good performance in a small-scale network. The linear

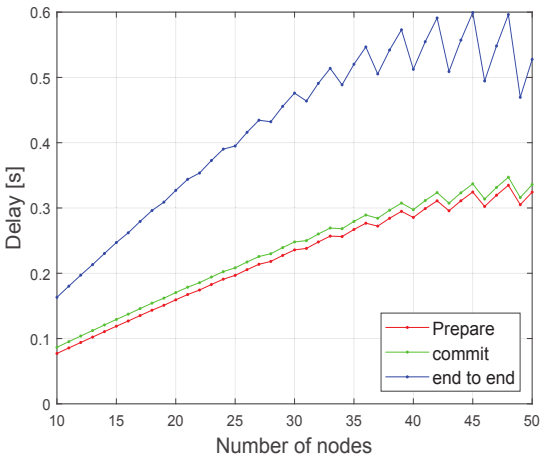

Fig. 8. Transaction confirmation delay versus number of nodes with $\lambda = 20$ and $W = 64$

expression of success probability, delay and throughput, along with the number of nodes, have been demonstrated. Even though a wireless PBFT network using IEEE 802.11 is tolerant to collisions and loss at a certain degree, the performance drops dramatically when the network scale reaches the threshold. Therefore, this paper provides precious guidance and instruction for the future wireless blockchain network construction deploying PBFT and IEEE 802.11.

References

[1] IBM Institute for Business Value, "Device democracy: Saving the future of the internet of things," https://www.ibm.com/services/us/gbs/thoughtleadership/internetofthings/, 2015, [Online; accessed 10-December-2020].

[2] M. Castro, B. Liskov *et al.*, "Practical byzantine fault tolerance," in *OSDI*, vol. 99, no. 1999, 1999, pp. 173–186.

[3] O. Onireti, L. Zhang, and M. A. Imran, "On the viable area of wireless practical byzantine fault tolerance (pbft) blockchain networks," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[4] L. Zhang, H. Xu, O. Onireti, M. A. Imran, and B. Cao, "How much communication resource is needed to run a wireless blockchain network?" *IEEE Network*, pp. 1–8, 2021.

[5] W. Li, C. Feng, L. Zhang, H. Xu, B. Cao, and M. A. Imran, "A scalable multi-layer pbft consensus for blockchain," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 5, pp. 1146–1160, 2020.

[6] B. Cao, M. Li, L. Zhang, Y. Li, and M. Peng, "How does csma/ca affect the performance and security in wireless blockchain networks," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4270–4280, 2020.

[7] F. Daneshgaran, M. Laddomada, F. Mesiti, and M. Mondin, "Unsaturated throughput analysis of ieee 802.11 in presence of non ideal transmission channel and capture effects," *IEEE transactions on Wireless Communications*, vol. 7, no. 4, pp. 1276–1286, 2008.

[8] J. C.-P. Wang, M. Abolhasan, D. R. Franklin, and F. Safaei, "Characterising the behaviour of ieee 802.11 broadcast transmissions in ad hoc wireless lans," in *2009 IEEE International Conference on Communications*, 2009, pp. 1–5.

[9] X. Ma and X. Chen, "Performance analysis of ieee 802.11 broadcast scheme in ad hoc wireless lans," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3757–3768, 2008.

[10] G. Wang, Y. Shu, L. Zhang, and O. W. Yang, "Delay analysis of the ieee 802.11 dcf," in *14th IEEE Proceedings on Personal, Indoor and Mobile Radio Communications, 2003. PIMRC 2003.*, vol. 2. IEEE, 2003, pp. 1737–1741.