IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received November 5, 2021, accepted December 9, 2021, date of publication December 24, 2021, date of current version January 6, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3138528

On the Use of Proof-of-Work in Permissioned Blockchains: Security and Fairness

IVAN MALAKHOV, ANDREA MARIN, (Senior Member, IEEE), SABINA ROSSI, AND DARIA SMUSEVA
DAIS, Università Ca' Foscari Venezia, 30172 Venice, Italy

Corresponding author: Sabina Rossi (sabina.rossi@unive.it)

The work of Sabina Rossi was supported in part by the INdAM - GNCS 2020 Project ''Sistemi Reversibili Concorrenti: dai Modelli ai Linguaggi.''

ABSTRACT In permissioned blockchains, a set of identifiable miners validates transactions and creates new blocks. In scholarship, the proposed solution for the consensus protocol is usually inspired by the Byzantine fault tolerance (BFT) based on voting rather than the proof-of-work (PoW). The advantage of PoW with respect to BFT is that it allows the final user to evaluate the cost required to change a confirmed transaction without the need to trust the consortium of miners. In this paper, we analyse the problems that arise from the application of PoW in permissioned blockchains. In standard PoW, it may be easy for colluded miners to temporarily reach 50% of the total hash power (HP). Moreover, since mining rewards are not usually expected in permissioned contexts, the problem of balancing the computational efforts among the miners becomes crucial. We propose a solution based on a sliding window algorithm to address these problems and analyse its effectiveness in terms of fairness and security. Furthermore, we present a quantitative, analytical model in order to assess its capacity to balance the hash power provided by heterogeneous miners. Our study considers the trade-off between the need to trust the entire consortium of miners guaranteed by the global HP invested by the mining process and the need to prevent collusion among malicious miners aimed at reaching 50% of the total HP. As a result, the model can be used to find the optimal parameters for the sliding window protocol.

INDEX TERMS Permissioned blockchain, Markov models, security, fairness.

I. INTRODUCTION

Blockchain technology has a great impact on current life, and it seems to be here to stay. Since the introduction of the first blockchain protocol in 2009, namely, Bitcoin [1], the popularity of this technology has grown dramatically. More people started joining the peer-to-peer (P2P) network underlying Bitcoin, giving rise to the golden age of cryptocurrencies.

However, since 2009, blockchain technology has been applied in areas other cryptocurrencies. Indeed, whenever one desires to store immutable data in a P2P network and needs a decentralized way to validate those data, a blockchain is a viable technology. Examples of applications based on blockchain technology are financial services, integrity verification, health care management, supply chain management, and others (see, e.g., [2], [3]).

A blockchain is a decentralized distributed network with an immutable and time-ordered ledger that contains records stored in blocks. Blocks are typically added to the blockchain by using the decentralized consensus mechanism named the *proof-of-work* (PoW), which behaves as follows. Users of a blockchain send data to be stored in the system (*transactions*), and these are fetched by special users called *miners* who try to consolidate them, i.e., add them to the blockchain in a permanent and unmodifiable way. Once this happens, the transaction is *confirmed*.

Miners compete to consolidate the block they are working on, and this competition favours those who possess high computational power while maintaining a certain degree of randomness. Since PoW is based on the computation of hash functions, we usually measure the computational power of miners as the number of hashes per unit of time that this miner is able to provide (*hash power* (HP)).

Another role of miners is to verify the validity of transactions added to the blocks and keep a local copy of the entire chain (or, at least, the most recent segment).

The associate editor coordinating the review of this manuscript and approving it for publication was Marco Anisetti.

Blockchains can be entirely public, as with Bitcoin. In this context, miners can be arbitrary users whose identity is, generally, unknown. These blockchains are called *permissionless*. However, there are other situations in which although the information stored in the ledger may still be publicly available, the miners belong to a set of well-known organizations. These blockchains are called *permissioned*. For example, if a set of companies decides to use its own blockchain to control and monitor a supply chain, then the miners are well known, although we still want to maintain P2P relations between them so that one cannot take control of stored data.

A. MOTIVATIONS

Most research papers studying permissioned networks consider consensus methods different from PoW. This is justified by at least two reasons. The first reason is that some scholars prefer to achieve a consensus with algorithms that are less computationally demanding with respect to PoW and with better scalability. The second reason is that in the case of PoW, it would be relatively easy for one of a group of colluding miners to take control of a blockchain by increasing the corresponding HP, e.g., by hiring new machines for a limited amount of time.

However, the motivations for discarding PoW from permissioned networks are still not obvious. Indeed, consider the most popular family of consensus algorithms for permissioned blockchains, the *Byzantine fault tolerance* (BFT) based on voting, i.e., those algorithms in which the consolidation process occurs once the majority of authorized miners (*committee*) validate the new block [4]. Hereafter, we refer to this class of algorithms by simply using the acronym BFT. On the one hand, BFT protects authorized miners from the malicious behaviours of a minority of them (what we call the *internal trust problem*). On the other hand, BFT does not necessarily guarantee the integrity of the data to the end users (the *external trust problem*).

Indeed, unlike what occurs in the presence of PoW, whenever there is an agreement among the miners of a committee, it is relatively easy for them to change any transaction stored in the blockchain since all the blocks following that with the modified transaction can be instantly regenerated upon agreement of the majority of the miners.

Thus, we can highlight two types of trust relations:

- The trust among the miners in one network (internal), and
- The trust of the end user on the miner pool (external).

Traditional PoW is, generally, unable to handle the internal trust problem. In fact, some miners could try to gain sufficient computational power in order to obtain an extra advantage and, in the worst case, to obtain control on the network. They would then be able to modify the entire blockchain history. Conversely, BFT is effective in ensuring internal trust.

Regarding the external trust problem, we have the opposite situation. In fact, PoW, as a different solution of the Byzantine generals problem proposed by Nakomoto, guarantees the expected cost of modifying a confirmed transaction since the HP used to ensure the immutability of data is publicly known and any change comes at the cost of computational power *by design*. BFT is unable to provide the same guarantees. Whenever the majority of miners agree on the modification of a transaction, they can implement out at basically no cost; hence, the end users must trust the consortium as a reliable entity.

To provide a better understanding, let us consider a case of a set of companies that maintain a blockchain to store publicly available pollution level data on a certain region in which they operate. Clearly, there is a first level of trust that concerns only the companies participating in the consortium. This can be easily achieved with the voting consensus mechanism. However, at the same time, citizens need to know that the historical data stored in the chain have not been changed; and whether they had been changed, a (possibly high) cost has been paid. BFT does not guarantee this since all companies may be interested in altering past data. Thus, PoW guarantees transparency for the end users, since any change in past data will require the computation of the hashes for the block containing the modified transactions and all the following blocks.

Another argument in favour of PoW is that although the classic PoW is vulnerable to 50% attacks, the BFT by voting is known to tolerate at most $\lfloor (n-1)/3 \rfloor$ of malicious nodes in a network with weak synchrony conditions [5]. For some cases in permissioned blockchains, it is easier for an attacker to find agreement among other miners than to obtain considerable computational resources as it comes at a cost.

Moreover, to maintain agreement among miners, a network with the BFT consensus mechanism by voting has to use synchronous communications while a PoW blockchain only relies on the timestamps of executing machines [6].

The protocol we propose gives quantitative guarantees for both internal and external trust problems.

Another important aspect of traditional PoW in permissioned networks concerns the balancing of work. In such blockchains, miners do not usually receive a reward for their mining; hence, selfish behaviour induces them to reduce the exposed HP with the aim of reducing the energy costs. As a consequence, a different mechanism with respect to rewarding must be adopted to even the HP used by miners. Notice that this notion of fairness is quite different from that of permissionless networks, where fairness is defined in such a way that the proportion of blocks (and hence the proportion of rewards) obtained by a miner is close to the proportion of his HP [7].

We study the impact of our solution in this regard, showing that an important positive side effect, our modified PoW fairly distributes the computational efforts among the miners of the consortium.

B. CONTRIBUTION

In this paper, we propose a simple PoW mining algorithm for permissioned networks, which is based on the use of a sliding window. The main idea is that each miner maintains a control window of size N that stores the information about the consolidators of the latest N blocks in the blockchain. The rule is that a miner m can be present in the window at most once. When a node receives a block from miner m and m is not present in the window, then it behaves as usual (i.e., it verifies the transactions and the hash; and if these are correct, it accepts the new block). Otherwise, the block is rejected. We will discuss how this approach addresses the internal and external trust problems mentioned above.

We study the security of this protocol with respect to the two major security threats of PoW: the 50% attack, which is particularly dangerous in permissioned networks; and the greedy miner attack in which a miner aims to consolidate more blocks than what is expected from the corresponding HP. Moreover, we provide a quantitative Markovian model of the system to study its fairness, which is here intended as the capability of reducing the gaps among the available HP of the miners.

We observe that these results pose an interesting trade-off. In fact, the total HP of the network is not used because the miners that are present in the window will stop their work; hence, external trust is quantitatively lower than that guaranteed by a plain PoW (which is, however, unable to guarantee internal trust). On the positive side, the HP that is unused, although available, does not become wasted energy. Thus, larger window sizes ensure a high internal trust by protecting the system from the collusion of miners; however, on the other hand, miners reduce the total HP devoted to guarantee the external trust, and vice versa. The quantitative model that we propose allows us to study this trade-off and determine the optimal configuration according to the design needs.

This paper is an extended and revised version of [8]. With respect to the conference version, in this paper, we consider the case of colluding miners. This scenario has relevant practical importance and requires nontrivial considerations in the model analysis.

C. STRUCTURE OF THE PAPER

Section II discusses some related work. In Section III, we present the window control blockchain protocol. In Section IV, we describe the Markvoian model for the performance evaluation of this protocol and give the algorithm for the computation of the relevant indices. In Section V, we analyse how the window control algorithm reacts to potential security attacks to permissioned networks. In Section VI, we analyse the impact of the window control on the fairness of the blockchain network. Finally, Section VII concludes the paper.

II. BACKGROUND AND RELATED WORK

Since the pioneering work by Nakamoto [1], many research efforts have been devoted to the analysis of the security and performance of blockchain systems. For example, in [9], the authors study how data broadcasting delays may favour certain types of attacks and unnaturally increase the number of forks. The importance of the contribution also relies on the network model and its configuration, which helps the understanding of blockchain fork occurrences.

Many works study the pros and cons of several consensus mechanisms. BFT or hybrid consensus mechanisms are proposed in [10], [11], [12], [13] and [14]. In some cases, BFT is also suggested for public blockchains. The main advantage of adopting a non-PoW consensus is the reduction of the power consumption and the greater transaction throughput. However, these systems cannot be considered to be as quantitatively secure as PoWs, where the end user can quantify the cost of a modification of a consolidated transaction.

Particularly, the authors [14] propose a ''new generation'' hybrid Bitcoin protocol where the process is separated on *leader election* and *transaction serialization* applying better scaling and throughput with a certain level of fairness. Although it helps to outperform the classical Bitcoin protocol, this protocol is still vulnerable to double spending and selfish miner attacks when the portion of the Byzantine nodes reaches at least 25% of all nodes.

In addition, one of the HyperLedger projects, Sawtooth, started supporting PBFT as a consensus approach in addition to the initially utilised *proof of elapsed time* mechanism. However, it is worth mentioning that the current versions of Sawtooth with PBFT are still recent implementations and have some limitations, such as the full peering requirement and lack of open network enrolment [15].

A comparison between PoW and BFT consolidation policies for permissioned blockchains was conducted by Vukolić in [6] and [16]. He analyses the issues of applying such consensus approaches predominantly in the permissioned settings. Finally, the author confirms that although BFT-based consensus mechanisms have the advantages of higher transaction rates and low energy consumption, PoW blockchain networks yield unique security features that we discuss in this paper.

Fairness is another property that attracts much attention from the community. In public networks, fairness is defined as the property that allows a miner to consolidate a fraction of blocks that is proportional to his fraction of HP. Note that this differs from our notion of fairness since we consider permissioned blockchain systems in which rewards for block mining are usually absent.

For public systems, [7] proposes a PoW-based protocol called the fruitchain. The authors prove that this protocol also achieves quantitatively predictable fairness in the presence of greedy miners. Note that unlike our proposal, fruitchain aims to obtain a network in which miners receive rewards that are proportional to the invested HP. In our case, we want to avoid the possibility that a miner controls a network by exposing a huge HP. In our sense, fairness means regulating the used HP of the miners so that they tend to consolidate the same number of blocks in the long run.

In [17], the authors study the fairness properties of a blockchain and describe the behaviour of two honest miners experiencing different propagation delays. The main result is that the propagation delay, as well as the HP, impacts the network fairness. Moreover, in the case of two miners with the same HP, there is an advantage for the miner who belongs to the stronger cluster of miners.

They demonstrate that as network latency increases, the protocol remains stable.

Fairness is addressed also in [18]. The authors introduce a new blockchain protocol called *DECOR+HOP*. It provides fairness among miners by distributing the block generation rewards among all the miners that originate the same forks. In this way, the overall fairness of the network is improved, and the expected number of forks is reduced.

Fairness in permissioned blockchains implementing proof-of-stake consolidation was investigated in [19]. In this context, fairness is defined in terms of distribution in the selection mechanism (*forming a committee*) and reward mechanism (*sharing goods*). The authors examine the fairness in synchronous systems and prove that it is the optimal solution.

With respect to our work, the latter two papers consider a completely different consolidation mechanism, i.e., proof-of-stake, while we rely on PoW.

III. THE PROBLEM STATEMENT AND WINDOW-BASED CONTROL

In permissioned networks, we distinguish the problem of trust among the miners and between the set of miners and public observers. While the latter is intrinsically guaranteed by PoW in the sense that even if all the miners agree on modifying a consolidated transaction they have to spend an amount of energy that is publicly known, the former problem requires more attention.

A. SECURITY VULNERABILITIES OF PoW

Here, we give a brief description of the security threats that affect permissioned blockchain networks:

- *50% attack*. A malicious miner can modify consolidated transactions when it controls at least 50% of the entire computational power of the network [20]. While this attack seems to be very unlikely in public chains with many miners, such as Bitcoin, in case of a restricted pool of miners, this may turn to be a serious threat to the security of the ledger. Indeed, it is possible for one or a small subset of the miners to hire a sufficient amount of computational power so that it can reach the 50% needed to violate the network security. However, in order to conclude the attack successfully, the malicious miner must be able to generate a number of consecutive blocks that coincide with the number of blocks that have been added after the modified block, plus the corrupted block itself.
- *Greedy miner attack*. A malicious miner that controls an amount of HP lower than 50% can consolidate a number of blocks that is still higher than the proportion of the corresponding HP, and this can be done in the following way. Once the attacker mines a new block, it does not immediately propagate the block. Instead, it keeps the block unannounced and announces it as soon as some other miner does so with a new block [21]. Although this problem mainly affects permissionless blockchains in which the rewards per block are expected, such as in Bitcoin, it may also be a problem in permissioned networks. Indeed, the creation of ‘unnatural’ chain forks may reduce the amount of total HP adopted to guarantee the immutability of the blockchain or, alternatively, may affect the overall system throughput.

B. FAIRNESS ISSUES IN PERMISSIONED BLOCKCHAINS

In blockchains, fairness is the property that distributes the efforts required by the distributed ledger evenly among the miners. However, the application of this principle is different for permissioned and permissionless blockchains, especially because the latter have a reward policy to incentive miners' efforts.

Indeed, the notion of fairness in permissionless blockchains is usually concerned with the fraction of rewards (or consolidated blocks) that a miner possessing a certain HP should statistically receive. This problem has been widely studied in [7] where an entirely new method, called the *fruitchain*, for consolidating blocks has been proposed.

In the present work, since permissioned networks do not usually adopt a reward mechanism, we propose a different notion of fairness. Ideally, given a certain level of mining difficulty, fairness is achieved when all the miners invest the same amount of HP to the life of the ledger. This is quite difficult to realize since different hardware can be used by miners.

If miners provide different HP, this means that they invest different energy resources (and hence financial efforts) to run the ledger while they all obtain the same service.

The trivial solution could consist of developing a round-robin scheme such that miners consolidate the blocks in turn. However, this solution is ineffective under our assumptions. In practice, the round-robin scheme would allow a miner to totally block the mining process either because of a fault or because of a malicious aim.

C. ALGORITHM DESCRIPTION: WINDOW-BASED CONTROL

In this paper, we propose a solution to the previously described problems based on a sliding window control algorithm. In this section, we formally describe the algorithm; and in Section IV, we describe its analysis.

Let us denote the set of M miners that are assumed to have their own identity as $\mathcal{K} = \{m_1, m_2, \ldots, m_M\}$. This means that miners are not anonymous and cannot consolidate new blocks under a different identifier, as could happen in permissionless systems.

Each miner maintains a control window of size N where it stores the identifiers of the latest N blocks' creators. At any time, in the window, at most one block of a miner $m \in \mathcal{K}$ can appear. Upon the announcement of a new mined block, one of these situations may arise:

- If the consolidator of the new block's identifier is already present in the window, then the miners will discard the new block, which is considered an invalid block.
- Otherwise, the block is considered valid under the assumption that all the other conditions are satisfied, e.g., it contains valid transactions and the PoW is solved correctly. The control window is updated with the new miner identifier according to a first-in-first-out (FIFO) policy.

It is worth noting that if there are no forks in the blockchain, then all the miners share the same control window. Otherwise, whenever a fork is solved, the control window must be updated coherently.

Observe that miners whose blocks would be rejected do not even participate in PoW competition. As a consequence, the total hash power that is used by the network is reduced; hence, the electric power consumed by the network also decreases. In Sections V and VI, we will study how this control mechanism may affect network behaviour in terms of security and fairness.

Notice that the protocol has two limiting cases. The first limiting case occurs when $N = 0$, i.e., the window mechanism control does not prevent any miners from adding new blocks to the chain. In this case, we obtain the standard PoW-based protocol. The other limiting case occurs when $N = M - 1$. In this case, the mining process follows a round-robin policy in which the blocks are consolidated by the miners in turn. The dimension of the window size allows the definition of intermediate operating conditions, and we will show that it prevents a single miner from taking control of the network (even with more than 50% of the computational power) while it secures the consolidated information with the well-known properties of the PoW algorithm. This tension between the need for a large window size to encourage fair involvement of all miners in the block consolidation process and the need for a small window size to exploit the PoW properties makes the model presented in Section IV crucial for a correct parameterization of the protocol and understanding of its security properties.

IV. A STOCHASTIC MODEL

In this section, we introduce a stochastic model that is based on continuous time Markov chains (CTMCs) for the sliding window control algorithm described in Section III.

The Markov chain underlying the model is ρ-reversible, as described in [22], [23]; hence, this guarantees high numerical tractability. This property allows us to use it to parameterize the model by setting appropriate window sizes. The model considers a single window and is subject to the following assumptions:

- Blocks are generated according to a Poisson process whose rate may depend on the state of the sliding window. This assumption is justified by the following argument. This process is generated by the superposition of the mining processes of all the miners. Indeed, it is well-known that the PoW requires the computation of a hash and this operation is memoryless. Hence, the time to the next block consolidation for miner m can be assumed to be exponentially distributed with a rate that is proportional to its hash power and that depends on the difficulty parameter set by the network. Moreover, since we can assume that the mining processes are independent, the block generating process is a Poisson process.
- In permissioned blockchains, forks are much rarer than in permissionless networks; therefore, the analyses that we can perform with our model can safely ignore the forks. As a consequence, in our analysis, we assume that there are no forks; hence, all the miners share the same control window.

We denote the window size as N and assume that $N < M$. At each epoch, the state of the window is denoted by vector $\vec{x} = (x_1, x_2, \ldots, x_N)$, where $x_i \in \mathcal{K}$. Moreover, we assume $|\vec{x}|_m = \sum_{i=1}^{N} \delta_{x_i=m}$ be 1 if m is in $\vec{x}$ and 0 otherwise. Finally, as described above, individual miner block generation is assumed to occur according to an independent Poisson process with rate λ_m.

Clearly, the stochastic process $X(t)$ underlying the temporal evolution of $\vec{x}$ is a homogeneous continuous-time Markov chain with finite state space. The transition rates of the CTMC infinitesimal generator are as follows: for $\vec{x}$ and $\vec{x}'$ such that $\vec{x} \neq \vec{x}'$,

$$q\left(\vec{x}, \vec{x}'\right) = \begin{cases} \lambda_m & \text{if } |\vec{x}|_m = 0 \text{ and} \\ & \vec{x}' = (m, x_1, \ldots, x_{N-1}) \\ 0 & \text{otherwise.} \end{cases},$$

The state space of $X(t)$ is

$$\mathcal{S} = \{\vec{x} \in \mathcal{K}^N : |\vec{x}|_m \leq 1 \text{ for all } m \in \mathcal{K}\}.$$

The stationary distribution of $X(t)$ can be analytically derived following the lines of [24]. The following theorem provides the exact expression.

Theorem 1: The stationary distribution $\pi(\vec{x})$ of $X(t)$ is:

$$\pi(\vec{x}) = \frac{1}{G} \prod_{m \in \mathcal{K}} \lambda_m |\vec{x}|_m, \tag{1}$$

where $G = \sum_{\vec{x} \in \mathcal{S}} \prod_{m \in \mathcal{K}} \lambda_m |\vec{x}|_m$.

We are interested in real applications where the number of instances of a miner in the window are relevant rather than the order in which they appear.

Corollary 1 provides an analytical expression for such an aggregated stationary probability.

Corollary 1: Let $\mathbf{n} = (n_{m_1}, \ldots, n_{m_M})$ denote an aggregated state with $n_m \in \{0, 1\}$ for all $m \in \mathcal{K}$, and $\sum_{m\in\mathcal{K}} n_m = N$. Let

$$\mathcal{S}_{\mathcal{K},N} = \left\{ \mathbf{n} : \sum_{m\in\mathcal{K}} n_m = N \text{ and } n_m \in \{0,1\}\ \forall m \in \mathcal{K} \right\}$$

be the set of all aggregated states.

The stationary probability of observing the aggregated state $\mathbf{n}$ is:

$$\pi_A(\mathbf{n}) = \frac{1}{G} N! \prod_{m\in\mathcal{K}} \lambda_m n_m.$$

Hereafter, for a model consisting of a set of miners $\mathcal{K}$ and a window size N, we denote the normalizing constant as $G_{\mathcal{K},N}$. Lemma 1 provides the analytical expression for the stationary probability of finding a miner m in the window. Note that this is expressed in terms of the ratio of the normalizing constants of different models.

Lemma 1: The marginal stationary probability of observing one block of miner $m \in \mathcal{K}$ in the window is:

$$\pi^m_{\mathcal{K},N} = N\lambda_m \frac{G_{\mathcal{K}\setminus\{m\},N-1}}{G_{\mathcal{K},N}}$$

where $G_{\mathcal{K}\setminus\{m\},N-1}$ is the normalizing constant corresponding to a model without miner m and a window of size $N-1$.

The next corollary provides the analytical expression for the throughput of a miner. Note that this is defined as the expected number of mined blocks per unit of time.

Corollary 2: In the steady state, the throughput for a miner $m \in \mathcal{K}$ is given by:

$$\lambda^*_m = \lambda_m \frac{G_{\mathcal{K}\setminus\{m\},N}}{G_{\mathcal{K},N}}. \tag{2}$$

According to our window-based control algorithm, miners whose identifier is present in the window are not joining the mining process. Thus, miner m does not completely use the corresponding HP. Hence, we define the *effective HP* of miner m as the HP, which is on average devoted to the mining process. Formally, this corresponds exactly to λ^*_m under the convention of measuring the HP in the expected number of consolidated blocks per unit of time.

We can compute the normalizing constant by applying the polynomial convolution algorithm presented in [24].

V. SECURITY AND PERFORMANCE ASSESSMENT

In this section, we discuss how the window control algorithm reacts to potential attacks to permissioned networks.

A. DOUBLE SPENDING AND GREEDY MINER ATTACKS

From a security perspective, the main advantage of window-based network control is its resistance to attacks that require the consecutive generation of the blocks by a subset of malicious miners. We recall that in permissioned networks, these attacks are possible because we assume that there is a conflict of interests among the miners, where one (or a small subset) of the miners may be interested in changing information that was previously stored in the ledger.

Collusion among malicious miners is possible, and we will consider this possibility. In our evaluation, we assume that malicious miners may collaborate to achieve the same aim, even by sharing their HPs. In other words, a miner whose identifier is present in the window can temporarily transfer its HP to another malicious colluded miner entitled to generate a new block.

As for single-miner threats, 50% and greedy-miner attacks cannot be conducted with the sliding window algorithm. More precisely, we observe the following:

- As noted in Section III, the 50% attack can be a serious problem for permissioned networks based on PoW, which is also one of the reasons for the popularity of BFT in these cases.

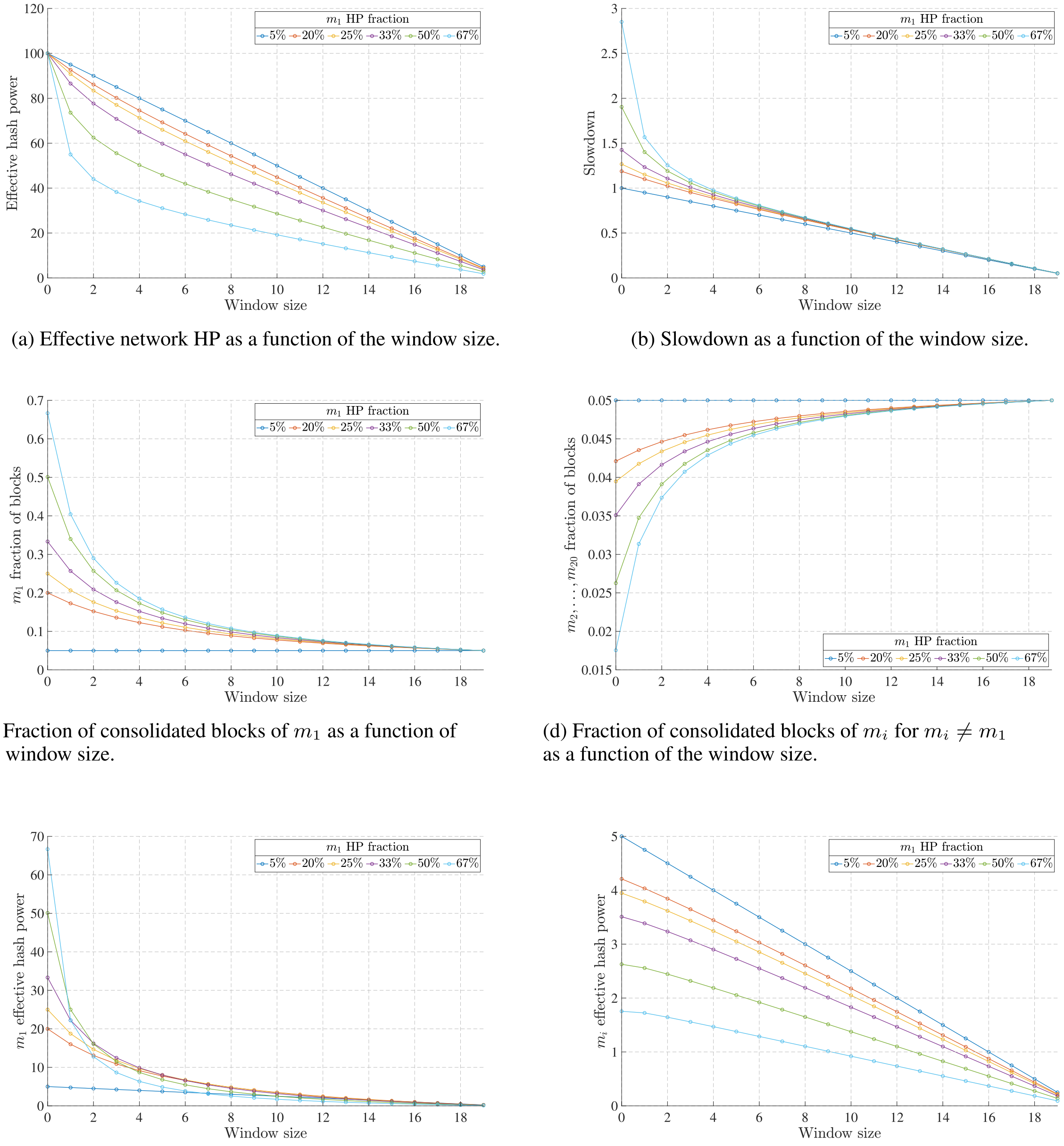
  The window control algorithm solves the problem as follows: if there is no collusion among the network miners, then the attacker must produce a certain number of consecutive blocks to conduct a 50% attack. Whenever the window size is positive, this is impossible because the other nodes would reject the proposed fork consisting of consecutive blocks consolidated by the same miner as invalid.
- For a successful greedy miner attack, the selfish miner needs to (i) produce blocks faster than other miners and (ii) make the fork accepted by the others. While the first phase is still doable in a window-controlled network, the second phase cannot be performed since the malicious miner must produce a longer chain of blocks than that actually in use to convince the remaining miners to accept his work. This would require him to mine consecutive blocks, which is again not allowed.

Consequently, it is clear that interested malicious miners will try to mitigate this crucial restriction in order to compromise the past data stored in a blockchain. One feasible solution that they could follow is finding the secret agreement with other miners. Thus, the above vulnerabilities will still occur in the case of several miners who will agree to cooperate. They will act as a mining pool in the network without the sliding window with the only difference that they will try to cheat and deceive others. In addition, if the window size is not smaller than the pool size, the colluded miners will be able to consecutively produce blocks as far as the size of the secret pool. Otherwise, if the window size is smaller than the secret pool size, then its block production is only limited by the fraction of their cumulative HPs.

B. SECURITY FOR A SINGLE MALICIOUS MINER

In this section, we consider the threat caused by a single miner that controls different amounts of HP. Recall that, in practice, this is achievable in permissioned networks rather easily since the computational power for the mining processes can be hired by the malicious miner.

From the functional point of view, the fact that the malicious miner (e.g., m_1) cannot consolidate consecutive



(a) Effective network HP as a function of the window size.

(b) Slowdown as a function of the window size.

(c) Fraction of consolidated blocks of m_1 as a function of the window size.

(d) Fraction of consolidated blocks of m_i for $m_i \neq m_1$ as a function of the window size.

(e) m_1 effective HP as a function of the window size.

(f) m_i effective HP for $m_i \neq m_1$ as a function of the window size.

FIGURE 1. Network with one unbalanced node.

blocks for any positive window size allows us to conclude that the protocol is safe for any fraction of the total HP controlled by m_1.

The impact of window control on the effective HP of the entire network λ_T^* remains to be assessed. Intuitively, this is the total expected HP of the miners that are not present in the window. This can be simply obtained by summing the effective HP of each miner as follows:

$$\lambda_T^* = \sum_{m \in \mathcal{K}} \lambda_m^*.$$

Recall that we measure the HP in terms of the expected number of blocks consolidated by a single miner in the unit of time under the condition that the network does not change the difficulty level of the PoW.

Let us consider the scenario in which 20 miners participate in the consolidation process, among which 19 are perfectly balanced, i.e., they expose the same HP. The remaining miner controls a variable fraction of HP that ranges from 5% to 67%. Formally, the vector of hash rates is the following:

$$\boldsymbol{\lambda} = \left(\lambda_1, \frac{100 - \lambda_1}{19}, \ldots, \frac{100 - \lambda_1}{19} \right).$$

Figure 1a shows the effective network HP λ_T^* as a function of the window size. Furthermore, the figure shows that there is a negative dependency between the network effective HP and the window size. On the one hand, larger window sizes result in a more balanced network; however, on the other hand, we have slower blockchain growth and this is where the trade-off between resource balance and performance appears. Note that if the PoW difficulty adapts to maintain a constant blockchain growth rate, as in the Bitcoin network, then we would compromise the PoW security by requiring a simpler hash computation. To clarify this point, let us consider the mostly unbalanced situation with which that we experimented, i.e., when miner m_1 has 67% of the total HP of the network. Clearly, the speed of the other miners is $(100 - 67)/19 = 1.74$. If m_1 could know this information, the ideal effective HP would be $1.74 \cdot 20 = 34.74$. With a window size of 1, Figure 1a shows that the effective HP is approximately 55. When moving further, the HP falls first to 44 and then gradually falls to almost 20 with a window size of 10. With even larger window sizes, the effective HP drops quickly to zero (with a window size of 20).

Figure 1b shows the slowdown of the window-based approach with respect to an ideal situation in which miners can agree to work at the speed of the slowest miner in a perfectly balanced way. In other words, the protocol reduces the number of blocks consolidated per unit of time in the attempt to achieve a fair condition. Since the miners do not explicitly agree on the HP, this is estimated by the use of the sliding window. In our case, the slowdown of the network's effective HP is defined as follows:

$$\mathcal{D} = \lambda_T^* \left(\frac{(100 - \lambda_{m_1}^*)}{M - 1} M \right)^{-1}.$$

Figure 1b suggests that for large window sizes, the slowdowns of the various scenarios tend to behave as the case of the fully balanced network. Indeed, starting from a window size of 10, the slowdown of every other network is very close to that of the system in which every miner controls 5% of the entire HP. This is explained by the fact that with a window size of 10, we already have a very well-balanced network; hence, the effects of window control on the system are almost indistinguishable from those observable in a perfectly balanced network.

It may be worth emphasizing the fact that when a miner is present in the window, it stops its mining process; therefore, the HP that is actually used by a node is in general smaller than the available one. Hence, the window control does not increase the energy wasted by the PoW.

Figures 1c and 1d show the percentages of nodes consolidated by miner 1 and the others, respectively. Notice that when the window size is 19, we have the round-robin discipline; hence, all the miners consolidate 5% of the blocks. We obtain the percentage of consolidated blocks by miner m_i as follows:

$$\frac{\lambda_{m_i}^*}{\sum_{m \in \mathcal{K}} \lambda_m^*}$$

i.e., this is proportional to the effective HP used by a miner. Figures 1c and 1d clearly show that small window sizes are sufficient to smooth out the gap between the HP of m_1 and the others. Indeed, larger window sizes have a strong impact on the effective HP while they give small benefits in terms of smoothing out the differences in miners' HPs. This can also be seen in the plots of Figures 1e and 1f. Specifically, note that while large windows have a negative effect on the effective HP of all miners, this is mostly evident for small window sizes and the unbalanced node m_1. For example, if we consider the case of one node controlling 67% of the total HP, his effective HP drops to 13 with a window size of 2.

To conclude, there are two effects of the window, even small windows, on a malicious miner hiring HP to overtake the other nodes: functionally, it prevents the mining of consecutive blocks; and quantitatively, it drastically reduces the imbalance created by this misbehaviour.

C. SECURITY ANALYSIS FOR POOLS OF COLLUDED MINERS

We analyse the case in which a subset of miners agree on cheating the protocol by changing a past transaction, resulting in a fork of the blockchain that gains the consensus of all the other miners.

Let $\mathcal{K}_C$ be the subset of colluded miners. There are three critical situations that should be considered:

1) $\mathcal{K}_C$ is a minority of all the miners,
2) $\mathcal{K}_C$ is a majority of all the miners that controls the minority of the HP, and
3) $\mathcal{K}_C$ is a majority of all the miners that controls the majority of the HP.

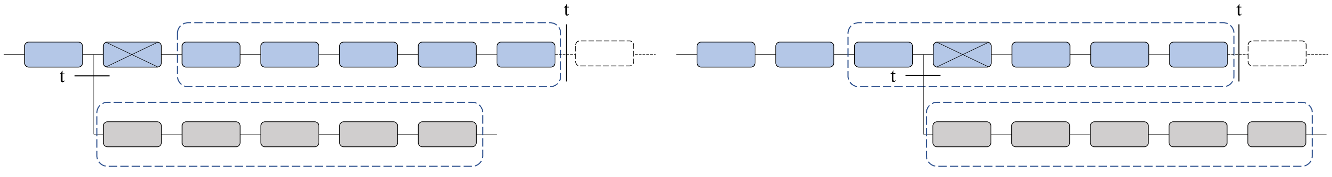
1) $\mathcal{K}_C$ is a minority of all the miners.

This scenario is a generalization of that considered in Section V-B. Although the system is robust to a single malicious miner, the possibility of collusion complicates the scenario and requires further investigation.

We have to consider two cases:

- $|\mathcal{K}_C| > N$. First, we distinguish the network where the number of colluded miners exceeds the window size. Since the colluded miners can transfer their HPs among each other and there is always at least one malicious miner who is not in the window, this case is equivalent to that of the 50% attack in an ordinary PoW blockchain.
- $|\mathcal{K}_C| \leq N$. First, we observe that the miners in $\mathcal{K}_C$ are unable to modify blocks that are more than $N - 1$ positions back in the chain, regardless of the percentage of the HP that they control. The blocks in the ledger older than the window size can be considered unmodifiable and hence safe with respect to such an attack.

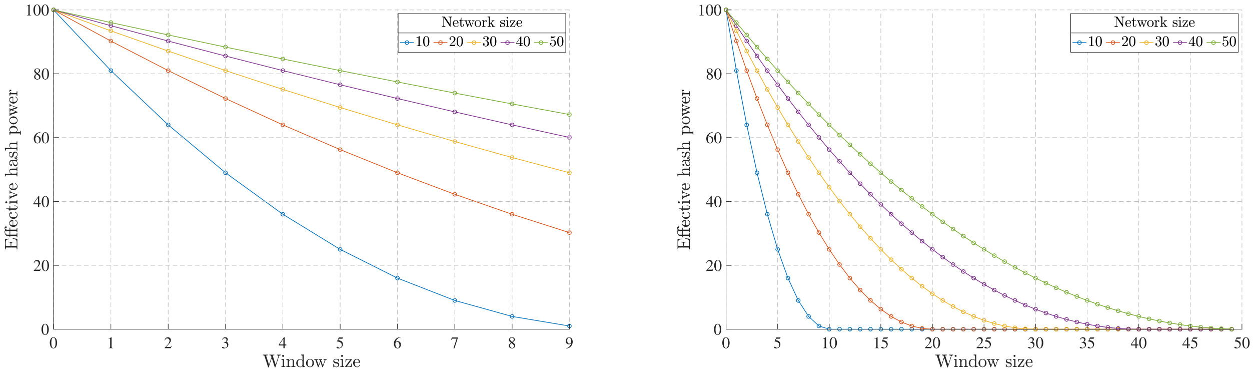
Example: As an example, consider the permissioned network with a window size of $N = 5$ and 11 miners, 5 of which are colluded and 6 are honest. Now, assume that a fraction $\alpha = 0.6$ of the total HP is controlled by the colluding miners. Therefore, $1 - \alpha = 0.4$ is the HP evenly distributed among the honest party as follows: $\frac{1-\alpha}{6}$. Since the miners in $\mathcal{K}_C$ can transfer their HPs among each other, the effective hash power of the malicious pool remains constant regardless



(a) Unsuccessful attempt to modify the target block.

(b) Successful attempt to modify the target block.

FIGURE 2. Demonstration of attempts to modify the past blocks of the blockchain of the colluded miner.



(a) Effective network HP as a function of the window size.

(b) Effective network HP as a function of the window size.

FIGURE 3. Different network sizes as the size of the window increases.

of the number of malicious miners out of the window. Figures 2a and 2b represent the cases where at time t the window contains five honest miners and the only remaining honest miner is available to create a new block. In the first example, we see that the colluded pool cannot change the past block marked with a cross. In fact, it is impossible for the malicious pool to create a longer chain than the existing chain because of the rules of the window. Figure 2b shows a successful attempt. Clearly, the effective HP of malicious miners is 9 times greater than the HP of the available honest miners. In terms of block creation, this means that by the time the honest miner has one block created, the miners in $\mathcal{K}_C$ will potentially have 9 blocks. However, the window size limits the actions of dishonest miners, and they can produce at most 5 consecutive blocks. It is clear that to have success, the colluded pool needs to overtake the honest party by creating the longest fork. Consequently, since it is impossible for them to produce more than N consecutive blocks in one row, any attempt to rewrite the blocks deeper or equal to N is impossible.

2) $\mathcal{K}_C$ is a majority of all the miners that controls the minority of the HP.

The second case is also worth investigation. We consider that all the honest miners have the same HP.

We first notice that if the window size is larger than the number of honest miners, then malicious miners can block the system by simply refraining from mining any new block. Therefore, henceforth, we assume that $N \leq |\mathcal{K}_H| < |\mathcal{K}_C|$, where $\mathcal{K}_H$ denotes the set of honest miners. In this case, the only protection against an attack is the difficulty experienced by the malicious pool to control the majority of the effective HP. In fact, recall that, for the sake of conducting a 50% attack, we need to consider the effective HP.

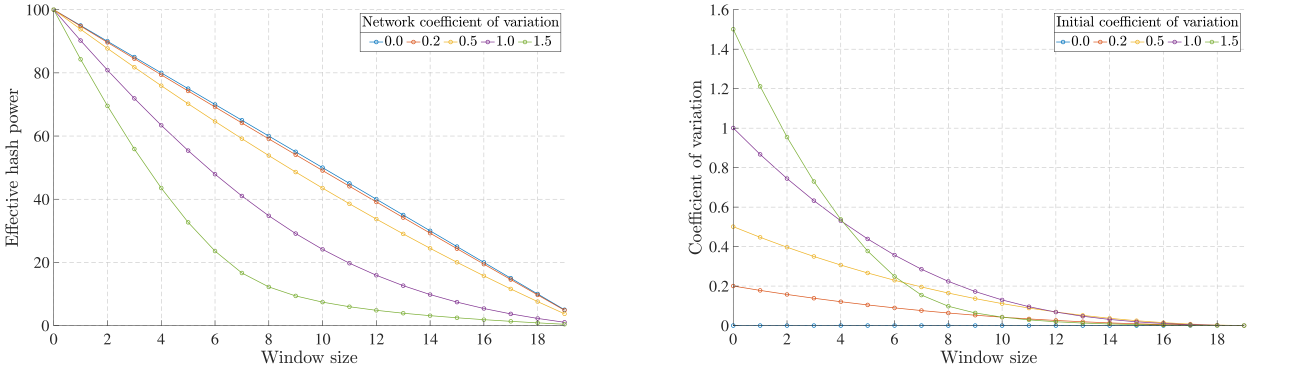
Suppose now that the colluded miners control HP $\lambda_C < \lambda_H$, where λ_H is the HP of the honest pool. Because of the assumption on the ability of colluded miners to transfer their computational power, we have $\lambda_C^* = \lambda_C$. Since $\mathcal{K}_C$ is working on a fork, only the remaining miners in $\mathcal{K}_H$ compete to access the window. Therefore, $\lambda_H^* < \lambda_H$. In other words, if the window size is too large, we can have the countereffect that we reduce the HP of honest miners too much, allowing malicious miners to succeed in a 50% attack. This observation emphasizes the importance of the proposed quantitative model to analyse the security trade-off that we just described.

To visualize the trade-off, we consider a scenario where the network of honest miners has sizes of 10, 20, 30, 40, and 50. For the sake of simplicity, we assume that they have the same HP. In addition, the window size is smaller than the number of colluded miners; thus, their HP coincides with their effective HP.

Figures 3a and 3b show the honest miner network's effective HP as a function of the window size. Consider, for example, a situation with 20 honest miners with an HP of 100 and 25 colluded miners that control an HP of 70. We see that the honest pool maintains the control of the majority of

TABLE 1. Miner hash power.

CV	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}	m_{11}	m_{12}	m_{13}	m_{14}	m_{15}	m_{16}	m_{17}	m_{18}	m_{19}	m_{20}
0.0	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000
0.2	4.2995	4.1846	6.0697	6.6619	4.3876	4.8923	4.0834	6.1523	4.8272	4.1176	4.3803	5.0134	4.1967	5.0965	4.1019	4.4555	4.3957	4.7517	6.5653	7.3668
0.5	9.3537	5.8357	1.3516	4.4557	2.0109	5.1976	2.4794	6.9398	3.3745	9.0610	3.4126	1.9169	2.4482	6.5069	4.1149	3.6289	8.1012	5.7067	5.3602	8.7433
1.0	11.1778	10.5201	0.2330	2.0742	1.8475	5.6199	0.3436	2.4887	0.4983	1.0483	15.8871	9.4988	0.4223	10.2551	0.3718	1.7351	4.1179	12.2617	8.5821	1.0168
1.5	0.7740	0.2101	0.3400	14.0988	7.6574	0.2508	18.2027	7.0961	0.3520	0.3511	0.1260	0.3471	22.4673	0.2870	6.9146	0.2040	0.2834	0.2387	19.4122	0.3865



(a) Effective network hash power as a function of the window size for different network coefficients of variation.

(b) Coefficient of variation of the miner HP fractions as a function of the window size for different network coefficients of variation.

FIGURE 4. Randomly generated scenarios.

the HP for window sizes up to 3. For larger window sizes, the security of the system can be compromised.

3) $\mathcal{K}_C$ is a majority of all the miners that controls the majority of the HP.

In this case, it is impossible to guarantee the security of the ledger as it can be intuitively understood. Indeed, for the same reasons described in the previous case, if $N \geq |\mathcal{K}_H|$, then the mining process can be blocked, while in the opposite case, the malicious pool trivially controls the majority of the effective HP and hence can succeed in a 50% attack. Note that in this situation, both PoW and voting-based agreement algorithms would be vulnerable.

VI. FAIRNESS ASSESSMENT

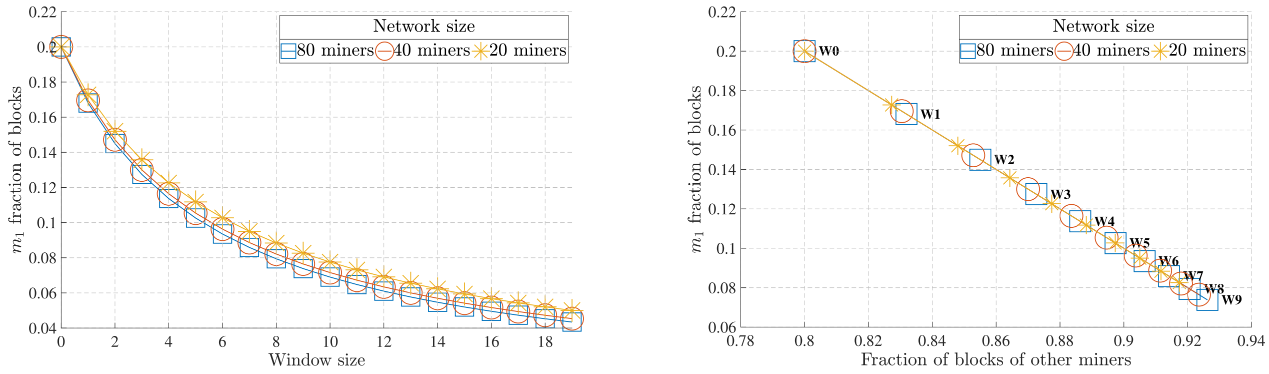
In this section, we analyse the impact of window control on the fairness of the blockchain network. Recall that, in this context, by fairness, we mean that every miner should invest the same amount of HP to secure the ledger. Therefore, we expect the effective HPs of miners to be closer to each other than their HPs.

Let us consider five scenarios where the HPs of the miners are randomly generated and shown in Table 1. For each scenario, we show the coefficient of variation (CV) of the distribution of miners' HP, where the first corresponds to the ideal case of a perfectly balanced network.

Figure 4a shows the network's effective HP as a function of the window size. We notice that, as expected, the window size negatively affects the effective HP, especially for networks with high dispersion of the miners' HP. In other words, if a consortium of miners finds an approximate agreement on the amount of HP to invest (which is clearly the practical scenario), then window control is able to smooth out the unavoidable differences caused by the impossibility of achieving a good balance with different hardware and the contingent situation that may face a server without reducing the effective HP too much.

Figure 4b shows the effect of window control on the coefficient of variation of the effective HPs λ_m^* used by miner m with $m \in \mathcal{K}$. These plots suggest that the small window sizes provide the highest benefits in terms of smoothing out the differences in the effective HPs of the miners. Let us focus on the scenario in which the coefficient of variation is 1. Note that with a window size of 2, the effective HP drops by 20% with respect to the maximum, and the coefficient of variation becomes 0.74. It may seem that the reduction of the HP is high, but that is not the case. Indeed, recall that even if the miners could find an agreement of the HP that they should use, we cannot raise the HP of the slowest, and hence that would become the speed of each individual miner. In the example, the slowest miner m_3 has an HP of 0.2330, which would lead to a total network HP of 4.66, which is much lower than the 80 obtained by window control. Clearly, there is a trade-off between the usability of the network and the need for fairness. From another point of view, window control is sufficiently flexible to allow miners some periods of small activity (or inactivity) without drastically reducing the effectiveness of the mining process.

Finally, we consider the network consisting of 19 balanced nodes and node m_1 controlling 20% of the HP of the entire network. We aim to show that the choice of the window size is very robust with respect to the number of miners. In other words, the effects that we observe with a certain window are very similar regardless of the number of nodes. This is shown



(a) Fraction of consolidated blocks of m_1 as a function of the window size.

(b) Fraction of the consolidated blocks of m_1 as a function of the fraction of the consolidated blocks by the others.

FIGURE 5. First scenario with different network sizes.

in Figures 5a and 5b. Figure 5a shows that with a window size of 0, node m_1 generates 20% of the blocks, as expected by its HP. With a window size of 3, this percentage is almost halved independent of the network size. Figure 5b confirms that the highest benefits in terms of controlling the system fairness are achieved with small window sizes; and to this aim, it seems useless to exceed a few units.

VII. CONCLUSION

In permissioned blockchains, we have to address two levels of security issues. The first concerns the possible lack of trust among the miners, and the second is the lack of trust between the end users and the consortium of miners.

In this paper, we have proposed a step towards the solution of the problems given by the application of PoW in permissioned blockchains. Indeed, although PoW is able to quantify the energy effort required to change a consolidated transaction, it is very weak in guaranteeing trust among miners. The sliding window control that we propose does not allow a miner to consolidate two or more blocks from the latest N blocks, where N is the window size. This does not allow a miner to take control of the mining process by reaching 50% of the total HP. Clearly, if this is a remote possibility for public blockchains such as Bitcoin, for permissioned networks, it represents a high risk.

Moreover, in permissioned blockchains adopting PoW, it is crucial to balance the effective HP provided by the miners. The idea behind this is reaching a certain level of fairness among the miners in terms of energy consumption. This is not a problem in systems where miners are rewarded for their work, but smoothing out the differences in the computational power of different miners in permissioned networks is important to encourage participation in the mining process.

The sliding window algorithm presented in this paper contributes to increase the security of PoW in permissioned blockchains and their fairness by reducing the effective HP used by the system. Indeed, miners whose id is present in the sliding window stop mining new blocks. This aspect of the algorithm clearly poses a trade-off problem that we addressed with the quantitative model.

The quantitative analysis that we conducted is based on a Markovian model and shows that small window sizes are sufficient to smooth out the differences in the potential HP of a heterogeneous group of miners, thus achieving fairness. Furthermore, the method is robust to the malfunctioning of some of the nodes that may be temporarily unavailable or faulty.

Finally, it is worth noting that its implementation requires minor changes to the existing PoW-based blockchain software and hence represents a viable solution to the abovementioned problems.

REFERENCES

[1] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” Tech. Rep., 2008.

[2] F. Casino, T. Dasaklis, and C. Patsakis, “A systematic literature review of blockchain-based applications: Current status, classification and open issues,” *Telematics Inform.*, vol. 36, pp. 55–81, Mar. 2019.

[3] Z. Zheng, S. Xie, H.-N. Dai, X. Chen, and H. Wang, “Blockchain challenges and opportunities: A survey,” *Int. J. Web Grid Services*, vol. 14, no. 4, pp. 352–375, 2018.

[4] W. Wang, D. T. Hoang, P. Hu, Z. Xiong, D. Niyato, P. Wang, Y. Wen, and D. I. Kim, “A survey on consensus mechanisms and mining strategy management in blockchain networks,” *IEEE Access*, vol. 7, pp. 22328–22370, 2018.

[5] M. Castro and B. Liskov, “Practical Byzantine fault tolerance,” in *Proc. 3rd Symp. Oper. Syst. Design Implement. (OSDI)*, 1999, pp. 173–186.

[6] M. Vukolić, “Rethinking permissioned blockchains,” in *Proc. ACM Workshop Blockchain, Cryptocurrencies Contracts*, New York, NY, USA, 2017, pp. 3–7.

[7] R. Pass and E. Shi, “FruitChains: A fair blockchain,” in *Proc. ACM Symp. Princ. Distrib. Comput. (PODC)*, E. M. Schiller and A. A. Schwarzmann, Eds., 2017, pp. 315–324.

[8] I. Malakhov, A. Marin, S. Rossi, and D. Smuseva, “Fair work distribution on permissioned blockchains: A mobile window based approach,” in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Nov. 2020, pp. 436–441.

[9] C. Decker and R. Wattenhofer, “Information propagation in the bitcoin network,” in *Proc. 13th IEEE Int. Conf. Peer-Peer Comput. (PP)*, Sep. 2013, pp. 1–10.

[10] A. Kiayias, A. Russell, B. David, and R. Oliynykov, “Ouroboros: A provably secure proof-of-stake blockchain protocol,” in *Proc. 37th Annu. Int. Cryptol. Conf.-Adv. Cryptol. (CRYPTO)*. Cham, Switzerland: Springer, 2017, pp. 357–388.

[11] I. Bentov, A. Gabizon, and A. Mizrahi, "Cryptocurrencies without proof of work," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Berlin, Germany: Springer, 2016, pp. 142–157.

[12] Z. Li, H. Wu, B. King, Z. B. Miled, J. Wassick, and J. Tazelaar, "A hybrid blockchain ledger for supply chain visibility," in *Proc. 17th Int. Symp. Parallel Distrib. Comput. (ISPDC)*, Jun. 2018, pp. 118–125.

[13] R. Pass and E. Shi, "Hybrid consensus: Efficient consensus in the permissionless model," in *Proc. 31st Int. Symp. Distrib. Comput. (DISC)*, vol. 91, 2017, pp. 39:1–39:16.

[14] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse, "Bitcoin-NG: A scalable blockchain protocol," in *Proc. 13th USENIX Symp. Netw. Syst. Design Implement. (NSDI)*, 2016, pp. 45–59.

[15] K. Olson, M. Bowman, J. Mitchell, S. Amundson, D. Middleton, and C. Montgomery. (2018). *Sawtooth: An Introduction*. [Online]. Available: https://www.hyperledger.org/wp-content/uploads/2018/01/Hyperledger_Sawtooth_WhitePaper.pdf

[16] M. Vukolić, "The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication," in *Prof. Int. Workshop Open Problems Netw. Secur.* Cham, Switzerland: Springer, 2016, pp. 112–125.

[17] R. Guerraoui and J. Wang, "On the unfairness of blockchain," in *Proc. Int. Conf. Netw. Syst. (NETYS)*. Cham, Switzerland: Springer, 2018, pp. 36–50.

[18] S. D. Lerner. (2015). *DECOR+HOP: A Scalable Blockchain Protocol*. [Online]. Available: https://scalingbitcoin.org/papers/DECOR-HOP.pdf

[19] Y. Amoussou-Guenou, A. Del Pozzo, M. Potop-Butucaru, and S. Tucci-Piergiovanni, "On fairness in committee-based blockchains," in *Proc. 2nd Int. Conf. Blockchain Econ., Secur. Protocols (Tokenomics)*, 2020, pp. 4:1–4:15.

[20] K. Chaudhary, V. Chand, and A. Fehnker, "Double-spending analysis of bitcoin," in *Proc. 24th Pacific Asia Conf. Inf. Syst. (PACIS)*, 2020, p. 210.

[21] A. Sapirshtein, Y. Sompolinsky, and A. Zohar, "Optimal selfish mining strategies in bitcoin," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Berlin, Germany: Springer, 2017, pp. 515–532.

[22] A. Marin and S. Rossi, "Autoreversibility: Exploiting symmetries in Markov chains," in *Proc. IEEE 21st Int. Symp. Modelling, Anal. Simulation Comput. Telecommun. Syst. (MASCOTS)*, Aug. 2013, pp. 151–160.

[23] A. Marin and S. Rossi, "On the relations between Markov chain lumpability and reversibility," *Acta Inf.*, vol. 54, no. 5, pp. 447–485, Aug. 2017.

[24] A. Marin, S. Rossi, D. Burato, A. Sina, and M. Sottana, "A product-form model for the performance evaluation of a bandwidth allocation strategy in WSNs," *ACM Trans. Model. Comput. Simul.*, vol. 28, no. 2, pp. 13:1–13:23, 2018.

IVAN MALAKHOV received the B.S. degree in information technology, information and communication and the M.S. degree in computer system networking and telecommunications, information and communication from the Higher School of Economics, Moscow, Russia, in 2017 and 2019, respectively, and the M.S. degree in computer science from the University Ca' Foscari of Venice, Italy, in 2019, where he is currently pursuing the Ph.D. degree in computer science. His research interests include the quantitative analysis of public and private blockchains.

ANDREA MARIN (Senior Member, IEEE) received the Ph.D. degree in computer science from the University Ca' Foscari of Venice, in 2007. He is currently an Associate Professor in computer science with the University Ca' Foscari of Venice. His research interests include the stochastic modeling of computer and communication systems for performance and reliability analysis, queuing theory, and models with product-form solutions. He has contributed to developing a probabilistic calculus for the formal analysis of wireless *ad hoc* networks.

SABINA ROSSI received the Ph.D. degree in computational mathematics and informatics from the University of Padova, in 1994. She has been a Visiting Professor with Universitè Paris 7, in 2007, and a Research Fellow with the Universitè Catholique de Louvain-la-Neuve, Belgium, in 1997. She is currently an Associate Professor in computer science with the University Ca' Foscari of Venice. Her current research interests include the development of formal tools for analysis and verification based on process algebraic techniques and, specifically, on stochastic process algebras.

DARIA SMUSEVA received the B.S. degree in information technology, information and communication and the M.S. degree in computer system networking and telecommunications, information and communication from the Higher School of Economics, Moscow, Russia, in 2017 and 2019, respectively, and the M.S. degree in computer science from the University Ca' Foscari of Venice, Italy, in 2019, where she is currently pursuing the Ph.D. degree in computer science. Her research interests include the quantitative analysis of public and private blockchains.