A Methodology for a Probabilistic Security Analysis of Sharding-Based Blockchain Protocols

Abdelatif Hafid[1], Abdelhakim Senhaji Hafid[2], and Mustapha Samih[1]

[1] Department of Mathematics, Faculty of Sciences, University Moulay Ismail, B.P. 11201 Zitoune Meknes, Morocco,
abdelatif.hafid@yahoo.com, samih.mustapha@yahoo.com
[2] Department of Computer Science and Operations Research, University of Montreal, Montreal, Canada,
ahafid@iro.umontreal.ca

Abstract. In the context of blockchain protocols, each node stores the entire state of the network and processes all transactions. This ensures high security, but limits scalability. Sharding is one of the most promising solutions to scale blockchain. In this paper, we analyse the security of three Sharding-based protocols using tail inequalities. The key contribution of our paper is to upper bound the failure probability for one committee and so for each epoch using tail inequalities for sums of bounded hypergeometric and binomial distributions. Two tail inequalities are used: Hoeffding [1] and Chvátal [2]. The first tail inequality Hoeffding [1] is much more precise bound. The second [2] is an exponential bound, it is simple to compute but weaker bound compared to Hoeffding [1]. Our contribution is an alternative solution when the failure probability simulations is impractical. To show the effectiveness of our analysis, we perform simulations of the exponential bound [2].

Keywords: failure probability, tail inequality, hypergeometric distribution, probabilistic security analysis, exponential bound

1 Introduction

Blockchain is a technology that, when used, can have a great impact in almost all industry segments including banking, healthcare, supply chain and government sector. It can be simply defined as a distributed digital ledger that keeps track of all the transactions (e.g. asset transfer, storage) that have taken place in a secure, chronological and immutable way using peer-to-peer networking technology. It does not rely on any trusted central entity (e.g. bank) to validate transactions and extend the blockchain; the network nodes (aka miners), using a consensus protocol, agree on which node can create (i.e. mine) a valid block and append it to the blockchain. For example, when Proof-of-work consensus protocol [3] is used, the node that first solves a mathematical puzzle, adds the block to the blockchain and gets rewarded (by the network and transaction fees). More specifically, a transaction is broadcasted to all the nodes in the network (1000 in the case of bitcoin); a node that receives the transaction, it checks whether the transaction is valid; if the response is yes, it sends the transaction to its neighbors; otherwise,

it drops the transaction. Periodically (e.g. each 10 minutes in Bitcoin [3]), a block (includes a list of transactions; e.g., up to 4000 transactions in Bitcoin) is created/mined and broadcasted to all the nodes in the network; the node who mined the block (first to solve the mathematical puzzle), appends the block to the blockchain and broadcasts it to its neighbors. A node that receives a block, it validates the block; if valid, it appends the block to the blockchain and broadcasts to its neighbors; otherwise, it drops it. Thus, in general, all nodes have the same copy of the blockchain; if not, nodes builds on the longest chain. One of the key limitations of proof-of-work based blockchains is scalability; indeed, the number of transactions that can be processed per second is small (e.g. up to 7 for Bitcoin and 15 for Ethereum [4]). This is unacceptable for most payment applications that require $1000s$ of transactions per second (e.g. Visa and PayPal). The objective of blockchain scalability is to process a high number of transactions per second (i.e. throughput) without sacrificing security and decentralization [5] [6]. Indeed, we can easily considerably increase the throughput but we will lose in terms of decentralization (wich is a key characteristic of blockchain).

A number of solutions to scale blockchain have been proposed; we can classify them into two categories: (1) On-chain solutions: they propose modifications to the blockchain protocols, such as Sharding (e.g. [7] [8] [9]) and block size increase (e.g. [10]); and (2) off-chain solutions (aka layer 2 solutions): these are built on the blockchain protocols; they process certain transactions (e.g. micro-payment transactions) outside the blockchain and only record important transactions (e.g. final balances) on the blockchain. Examples of layer 2 solutions include Lightning Network [11], Raiden Network [12], Plasma [13], and Atomic-swap [14]. Security and decentralization should be taken into account while solving the scalability issue in public blockchains. This is called the scalability trilemma; indeed, finding a balance between scalability, security and decentralization is very challenging. In this paper, we focus on analyzing the security of scalability solutions that use the concept of Sharding; this is motivated by the fact that Sharding is one of the promising solutions to the scalability problem. The basic idea behind Sharding is to divide the network into subsets, called shards; each shard will be working on different set of transactions rather than the entire network processing the same transactions. Several Sharding protocols have been proposed in the literature; they include Elastico [15], OmniLedger [16], RapidChain [17], Zilliga [18] and PolyChard [19]. Generally, Sharding is used in non-byzantine settings (e.g. [20]); Elastico [15] is the first Sharding-based protocol with the presence of byzantine adversaries. Elastico, divides the network into multiple committees where each committee handles a separate set of transactions, called shard. The number of shards grows nearly linearly with the size of the network. When the network grows up to $1,600$ nodes, Elastico succeeds at increasing the throughput (e.g. up to 40 transactions per second (tx/sec)). However, it has shortcomings that include: (1) the randomness used in each epoch (i.e in each fixed time period; e.g., once a week) of Elastico can be biased by malicious nodes; and (2) it can only tolerate up to 25% of malicious/faulty nodes (total resiliency) and 33% of malicious nodes in each committee (committee resiliency). OmniLedger [16] has been proposed to fix some of the shortcomings of Elastico. In particular, it uses a bias-resistant public-randomness protocol to ensure security. The OmniLedger consensus protocol uses a variant of ByzCoin [21] to handle and achieve faster transac-

tions (e.g. up 500 tx/sec when the network grows up to $1,800$ nodes). Omniledger, like Elastico claims the same resiliency for both; total resiliency and committee resiliency. Recently, Zamani and Movahedi in [17] proposed RapidChain as a Sharding-based public blockchain protocol which succeeds at outperforming existing Sharding algorithms (e.g. [15][16]) in terms of scalability and security. Indeed, RapidChain can tolerate up to 33% of malicious/faulty nodes and 50% of malicious nodes in each committee. RapidChain claims a high throughput (e.g. up to $4,220$ tx/sec when the network grows up to $1,800$ nodes). The table bellow summarizes common characteristics of related protocols used in our analysis. In this paper, we present a probabilistic security analysis of

Table 1: Resiliency Bound

Protocols	Total Resiliency	Committee Resiliency
Elastico [15] and Omniledger [16]	$\frac{1}{4}$	$\frac{1}{3}$
Rapidchain [17]	$\frac{1}{3}$	$\frac{1}{2}$

Elastico, OmniLedger and RapidChain. More specifically, we propose a probabilistic security analysis of these protocols using hypergeometric and binomial distributions. First, we calculate the failure probability for one committee; then, we calculate the union bound (i.e. the failure probability of each epoch); finally, we bound the failure probability with two bounds making use of the tail inequalities bounds [22] [2][1]. The first bound [1] is much more precise tail bound ; the second [2] is an exponential bound which is more simple and elegant bound, however weaker bound compared to [1]. Thereafter, we upper bound the failure probability for each epoch by multiplying the committees bounds by the number of committees.

The contribution of this paper consists of a solution to analyze security (i.e. computing failure probability bounds) when failure probability simulation is unpractical (e.g. required number of simulations increases as the number of shards increases). To the best of our knowledge, this is the first time that Hoeffding [1] and chvátal [2] inequalities are used to analyze security of blockchain protocols. We implemented the exponential bound function [2] in order to verify and show the effectiveness of our analysis.

The paper is organized as follows. Section 2 presents the proposed probabilistic analytical model. Section 3 evaluates the model. Finally, Section 4 concludes the paper.

2 Analytical Model

2.1 Notations

The table 2 shows the notations we used in the paper. Note that the cumulative hypergeometric distribution $H(K,N,n,k)$ is the sum for all $i \geq k$ of the probability distribution function $h(K,N,n,i)$.

Table 2: Useful Notations

Notation	Meaning
N	The total number of nodes
n	The committee size
K	The total number of malicious nodes
n_c	The number of committees
p_c	The committee failure probability
p_0	The bootstrap probability for RapidChain
$h(K,N,n,k)$	The hypergeometric distribution with parameters K, N and n
$H(K,N,n,k)$	The cumulative hypergeometric distribution with parameters K, N and n
$\mathrm{B}(n,p,k)$	The cumulative binomial distribution with parameters n and p
X	Random variable which represent the number of malicious nodes

2.2 Probability distributions

We use the hypergeometric and binomial distribution to calculate the failure probability for one committee and then for each epoch. We define the probability that a committee contains k malicious nodes sampled from a population of N nodes containing at most K corrupt nodes. Let X denote the random variable corresponding to the number of malicious nodes in the sampled committee. If we assume that X follows the hypergeometric distribution with parameters K, N and n, the failure probability is:

$$h(K,N,n,k) = \frac{\binom{K}{k}\binom{N-K}{n-k}}{\binom{N}{n}} \tag{1}$$

In this paper, we are interested in the probability that there is X, smaller than k malicious nodes when randomly selecting a committee of n nodes without replacement from a population of N nodes containing at most K corrupt nodes. The cumulative hypergeometric distribution function allows us to calculate this failure probability; indeed, the failure probability for one committee for Elastico and OmniLedger is:

$$H(K,N,n,\frac{n}{3}) = \sum_{k=\lfloor \frac{n}{3} \rfloor}^{n} \frac{\binom{K}{k}\binom{N-K}{n-k}}{\binom{N}{n}} \tag{2}$$

In general, when the hypergeometric distribution is used, a comparison is performed with the binomial distribution. More specifically, it is said that if n is small relative to the population size N, then X could be approximated by a binomial distribution. Practically, we approximate hypergeometric distribution by a binomial distribution when the sample size is smaller than 10% of the population [23]. However, when the sample size

gets larger relative to the population size, it is recommended to use the hypergeometric distribution (the hypergeometric distribution yields a better approximation in this case). If the sampling is done with replacement, we use the cumulative geometric distribution [22] or cumulative binomial distribution [23] instead of the cumulative hypegeometric distribution to calculate the failure probability. Now, if we assume that $X \sim \mathrm{B}(n,p)$ (i.e. X follows the binomial distribution with parameters n and p) where $p = \frac{K}{N}$, p is the probability of each node being malicious. Thus, the failure probability of one committee for Elastico and OmniLedger using the cumulative binomial distribution function is:

$$P(X \geq \frac{n}{3}) = \sum_{k=\lfloor \frac{n}{3} \rfloor}^{n} \binom{n}{k} p^k (1-p)^{n-k}. \tag{3}$$

2.3 Tail Inequalities

The main contribution of our work is to upper bound the failure probability for one committee and so for one epoch using two bounds functions. The tail inequalities are powerful results that can be compute these bounds [22][2][1]. Firstly, we upper bound the failure probability for one committee as well as for each epoch. The following bound is given by Hoeffding [1]:

$$H(K,N,n,k) \leq G(x), \tag{4}$$

where

$$G(x) = \left(\left(\frac{p}{p+x} \right)^{p+x} \left(\frac{1-p}{1-p-x} \right)^{1-p-x} \right)^{n}, \tag{5}$$

$p = \frac{K}{N}$ and $k = (p+x)n$ with $x \geq 0$.

Hence, we can bound the failure probability of one committee for Elastico and OmniLedger as follows:

$$H(K,N,n,\frac{n}{3}) \leq G(x), \tag{6}$$

where

$$x = \frac{1}{3} - p, \quad (p \leq \frac{1}{4}).$$

Likewise, we upper bound the failure probability of one committee for RapidChain:

$$H(K,N,n,\frac{n}{2}) \leq G(x), \tag{7}$$

where

$$x = \frac{1}{2} - p, \quad (p \leq \frac{1}{3}).$$

The binomial distribution coincidentally has an analogous tail bound [2], which means:

$$\mathrm{B}(n,p,k) \leq G(x), \tag{8}$$

where

$$\mathrm{B}(n,p,\frac{n}{2}) = \sum_{k=\lfloor \frac{n}{2} \rfloor}^{n} \binom{n}{k} p^k (1-p)^{n-k}.$$

Now, we upper bound the failure probability of each epoch for RapidChain; we calculate the union bound over n_c committees, where each committee can fail with probability p_c. When the sample size is smaller than 10%, p_c is calculated using cumulative binomial distribution. Otherwise, we use the cumulative hypergeometric distribution. In the first epoch for RapidChain protocol, the committee election procedure can fail with probability $p_0 = 2^{-26.36}$ (see [17]). Thus, the failure probability for one epoch for RapidChain is upper bounded as follows:

$$p_0 + n_c p_c \leq V(x), \tag{9}$$

where

$$V(x) = p_0 + n_c G(x), \qquad n_c = \frac{N}{n}.$$

Secondly, Chvátal [2] propose another tail bound, it is simple and elegant (i.e. exponential function), but weaker bound compared to the last one. We obtain the following bound:

$$H(K,N,n,k) \leq F(x), \tag{10}$$

where

$$F(x) = \exp^{-2x^2 n}.$$

Thus, the failure probability for one epoch for RapidChain is bounded as follows:

$$p_0 + n_c p_c \leq U(x), \tag{11}$$

where

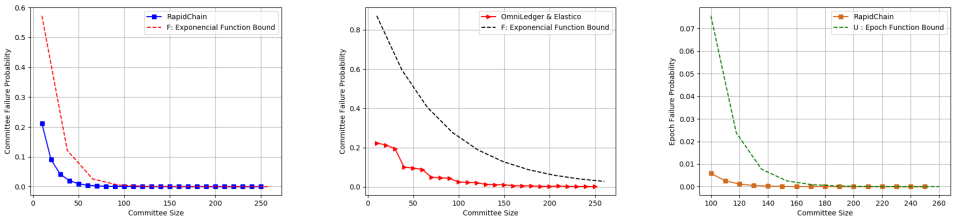
$$U(x) = p_0 + n_c F(x), \qquad n_c = \frac{N}{n}.$$

Similarly, we can upper bound the failure probability for each epoch for Elastico and OmniLedger.

3 Results and Analysis

Figure 1 shows the exponential tail bound and the probability of failure calculated using the hypergeometric and binomial distributions to sample a committee without replacement with various sizes from a pool of $2,000$ nodes. In particular, Fig. 1(a) shows the plot of the failure probability for one committee as well as the exponential function bound in RapidChain. We observe that the exponential bound curve looks similar to the curve of the failure probability calculated when the committee size increases (when it approaches 100). Hence, we get a good approximation bound when the committee size gets larger. Fig. 1 (b) shows the plot of the exponential tail bound of the failure probability for one committee in the OmniLedger and Elastico and the failure probability both decrease when the committee size increases; in addition, when the committee

size increases above 250 nodes, both curves look similar. Finally, Fig. 1 (c) presents the shows of the exponential bound of the failure probability for one epoch in RapidChain and the failure probability for the union bound over the number of committees while varying the committee size. We conclude that our proposal allows to compute bounds with good precision especially in the case of larger committee sizes.



(a) (b) (c)

Fig. 1: Plot of the exponential bounds, as well as the failure probability vs. the committee sizes; (a) for one committee for RapidChain [17], (b) for one committee for Elastico [15] and OmniLedger [16], and (c) for one epoch for RapidChain [17].

4 Conclusion and future works

In summary, we proposed two bounds of the failure probability for one committee, thereafter for each epoch when we use the hypergeometric or the binomial distribution using tail inequalities. The first bound is more precise, but difficult to compute. The second is a simple exponential bound whereas weaker bound compared to the last one. We also calculated the failure probability for one committee as well as for one epoch using hypergeometric and binomial distributions. We have approximated the hypergeometric distribution with the binomial distribution when the sample size smaller than 10%. We have implemented the exponential bound and the failure probability to show the performance of our analysis. We conclude that our proposal can be used to analyze security of any Sharding-based protocol. For the future work, we will apply tail bounds which are more precise and can yield good approximations. Another interesting work is to make a probabilistic security analysis of Ethereum-Sharding.

References

1. W. Hoeffding, “Probability inequalities for sums of bounded random variables,” in *The Collected Works of Wassily Hoeffding*. Springer, 1994, pp. 409–426.
2. V. Chvátal, “The tail of the hypergeometric distribution,” *Discrete Mathematics*, vol. 25, no. 3, pp. 285–287, 1979.
3. S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008.

4. G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum project yellow paper*, vol. 151, pp. 1–32, 2014.
5. bloXroute Team, "The scalability problem, (very) simply explained," 2018.
6. BitRewards, "Blockchain scalability: The issues, and proposed solutions," 2018.
7. Z. Ramy, "Here's the deal on sharding," 2018.
8. H.-W. Wang, "Ethereum sharding: Overview and finality," 2017.
9. A. E. Gencer, R. van Renesse, and E. G. Sirer, "Short paper: Service-oriented sharding for blockchains," in *International Conference on Financial Cryptography and Data Security*. Springer, 2017, pp. 393–401.
10. J. Garzik, "Block size increase to 2mb," *Bitcoin Improvement Proposal*, vol. 102, 2015.
11. J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," 2016.
12. R. Network-Fast, "cheap, scalable token transfers for ethereum," 2018.
13. J. Poon and V. Buterin, "Plasma: Scalable autonomous smart contracts," *White paper*, pp. 1–47, 2017.
14. Komodo, "Advanced blockchain technology, focused on freedom," 2018.
15. L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A secure sharding protocol for open blockchains," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 17–30.
16. 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)*. IEEE, 2018, pp. 583–598.
17. M. Zamani, M. Movahedi, and M. Raykova, "Rapidchain: scaling blockchain via full sharding," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2018, pp. 931–948.
18. Z. Team *et al.*, "The zilliqa technical whitepaper," 2017.
19. S. Li, M. Yu, S. Avestimehr, S. Kannan, and P. Viswanath, "Polyshard: Coded sharding achieves linearly scaling efficiency and security simultaneously," *arXiv preprint arXiv:1809.10361*, 2018.
20. G. Danezis and S. Meiklejohn, "Centrally banked cryptocurrencies," *arXiv preprint arXiv:1505.06895*, 2015.
21. E. K. Kogias, P. Jovanovic, N. Gailly, I. Khoffi, L. Gasser, and B. Ford, "Enhancing bitcoin security and performance with strong consistency via collective signing," in *25th {USENIX} Security Symposium ({USENIX} Security 16)*, 2016, pp. 279–296.
22. M. Skala, "Hypergeometric tail inequalities: ending the insanity," *arXiv preprint arXiv:1311.5939*, 2013.
23. J. Wroughton and T. Cole, "Distinguishing between binomial, hypergeometric and negative binomial distributions," *Journal of Statistics Education*, vol. 21, no. 1, 2013.