Probability in the Engineering and Informational Sciences, **35**, 2021, 858–866.
doi:10.1017/S0269964820000212

BLOCKCHAIN DOUBLE-SPEND ATTACK DURATION

MARK BROWN
Department of Statistics, Columbia University, New York, NY, USA
E-mail: pekoz@bu.edu

EROL PEKÖZ
School of Business, Boston University, Boston, MA, USA

SHELDON ROSS
Department of Industrial and Systems Engineering, University of Southern California, Los Angeles, CA, USA

Many cryptocurrencies including Bitcoin are susceptible to a so-called double-spend attack, where someone dishonestly attempts to reverse a recently confirmed transaction. The duration and likelihood of success of such an attack depends on the recency of the transaction and the computational power of the attacker, and these can be related to the distribution of time for counts from one Poisson process to exceed counts from another by some desired amount. We derive an exact expression for this distribution and show how it can be used to obtain efficient simulation estimators. We also give closed-form analytic approximations and illustrate their accuracy.

Keywords: applied probability, reliability theory

1. INTRODUCTION

Cryptocurrencies reached a total market value of nearly 1 trillion US dollars in 2018.[1] Part of their appeal is the reliability of the database, also called the blockchain, of confirmed transactions: once a transaction is confirmed, it is extremely difficult to reverse it. Transactions are not stored in a single central location but rather are stored locally by all participants. One primary innovation is the system of incentives in place to ensure this reliability is maintained, even without a central regulator (though Eyal and Sirer [3] points out a potential flaw in the incentive system).

A simplified setting for confirming the chain of transactions for Bitcoin, the largest cryptocurrency, is as follows. Miners start with the longest chain known so far and compete for a reward to be the first to discover a number, called the "nonce," that will

[1] See https://coinmarketcap.com/charts/.

© Cambridge University Press 2020 0269-9648/19 $25.00

https://doi.org/10.1017/S0269964820000212 Published online by Cambridge University Press

confirm the current block of pending transactions and extend the chain by incorporating these new transactions (see [10]) into a new block. The nonce discovery times for a given miner follows roughly a Poisson processes with the rate that depends on the computational power of the miner (see [9], though also see [2], who argue that a nonhomogeneous process fits the observed data better), and these Poisson processes are independent across miners. The reasonableness of the Poisson process as a model can be argued intuitively as follows. Miners exhaust a very large search space at random, and each trial is independent of the previous with roughly an equal but small probability of success. The number of trials until success, therefore, has a geometric distribution with a large mean and, since each trial is quick, the total time required is roughly a geometric with large mean multiplied by a very small constant, which is well approximated by the exponential distribution.

A well-known risk is the "double-spend" attack, where a dishonest miner with high computational power reverses a past transaction and tries to quickly re-confirm all the subsequent blocks; if the miner is able to surpass the collective honest miners and obtain a longer chain, the incentives will ensure that everyone accepts this chain going forward. To avoid the risk of this, the convention is to wait for a given number of subsequent confirmed blocks to be publicized before assuming that a transaction is secure. Historically this risk has been viewed a largely theoretical, but Orcutt [11] has recently pointed out that these attacks have been occurring more frequently now that miners can now easily rent a majority of the computational power to launch such attacks.

2. BACKGROUND AND RELATED WORK

There have been many prior papers studying double-spend and blockchain security-related issues. Pinzon and Rocha [14] studies the chance that the double-spend attack can occur under two different models and gives an overview of the various mining models. Ozisik and Levine [13] gives an analysis of the original mining model from [9]. Ozisik and Pinar [12] gives a method for reducing double-spend attacks and increasing transparency, and Saia *et al.* [17] gives an approach for increasing security of the blockchain, while Wang et al. [18] and Lin and Liao [8] survey the blockchain technology. Georgiadis and Zeilberger [4] and Rosenfeld [15] both study the double-spend success probability, and the latter also studies the duration moments. Goffard [5] also studies the duration and probability of a successful double-spend attack, but the formulas obtained there are difficult to evaluate; our formulas can be considered an improvement [6].

In this paper, we use a model to study the distribution of the duration of time it takes for double-spend attacks to succeed. Our model is as follows. Let $\{N_1(t), t \geq 0\}$ and $\{N_2(t), t \geq 0\}$ be independent Poisson processes with respective rates λ and μ. These correspond to the blocks created respectively by dishonest and honest miners. Also, let T_k denote the first time that the N_1 process is k greater than the N_2 process, or let it be ∞ if that never happens. That is, for $k > 0$,

$$T_k = \inf\{t \geq 0 : N_1(t) = N_2(t) + k\}.$$

Here, we study the distribution of T_k. This corresponds to the time until the dishonest miner can catch up to the honest miners when the honest miners have an initial lead of k mined blocks. Though there are many additional complexities of the mining process, such as gathering/validating transactions, finding the nonce, and then successfully broadcasting the block to the network, here we concentrate solely on nonce discover, for which our model is reasonable.

https://doi.org/10.1017/S0269964820000212 Published online by Cambridge University Press

The organization of the paper is as follows: in Section 3.1, we give an exact formula for the distribution and also give a method for bounding it and for simulating it efficiently, in Section 3.2, we give a closed-form Edgeworth-type approximation, and in Section 3.3, we illustrate the accuracy of this improved approximation compared with a normal approximation. In Section 4, we give a summary and conclusions.

Our model is a special case of one considered in Gofffard [5], where the density function of T_k is derived in a model that supposes that $\{N_2(t), t \geq 0\}$ is an order statistic point process (OSPP), which is a counting process $\{N(t)\}$ having the property that conditional on $N(t) = n$, the event times are distributed as the order statistics of a random sample from a distribution F_t (the nonhomogeneous Poisson process being an example of an OSPP), whereas $\{N_2(t)\}$ is allowed to be either a renewal process or an OSPP. In addition, the expression for the density of T_k is simplified in the important special case we are considering, where both counting processes are Poisson processes. However, even the simplified formula given in Gofffard [5] is different than the one we derive, being quite involved and not apparently useful either for obtaining numerical bounds or for simulation purposes.

3. PROPOSED APPROACH

3.1. Exact Results

To derive the distribution of T_k, we make use of a couple of lemmas. The first one, known as the ballot theorem, is well known and easily established by conditioning on the last element of the permutation.

LEMMA 1: *In a random permutation of n values $+1$ and m values -1, where $n > m$, the probability that for every $i = 1, \ldots, n+m$, the first i elements of the permutation always contain more values $+1$ than -1 is $(n-m)/(n+m)$.*

Now, consider the simple random walk $\{S_n, n \geq 1\}$, where $X_i, i \geq 1$, are independent and such that $P(X_i = 1) = p = 1 - P(X_i = -1)$ and $S_n = \sum_{i=1}^{n} X_i$. For $k > 0$, let

$$N_k = \min\{n : S_n = k\},$$

where $N_k = \infty$ if $S_n < k$ for all n.

LEMMA 2:

$$P(N_k = k + 2i) = \frac{k}{k+2i}\binom{k+2i}{k+i} p^{k+i}(1-p)^i, \quad i = 0, 1, \ldots$$

PROOF: Because $N_k = k + 2i \Rightarrow S_{k+2i} = k$, we have that

$$P(N_k = k + 2i) = P(S_{k+2i} = k) P(N_k = k + 2i | S_{k+2i} = k).$$

Now, $S_{k+2i} = k$ if $X_1, \ldots, X_{k+2i}$ consists of $k + i$ values $+1$ and i values -1. As $X_1, \ldots, X_n$ are exchangeable, it easily follows that, conditional on $S_{k+2i} = k$, $X_{k+2i}, \ldots, X_1$ is equally likely to be any of the possible orderings of $k + 2i$ values $+1$ and i values -1. Because an ordering will result in $S_1 < k, \ldots, S_{k+2i-1} < k$, if for all $j \leq k + 2i$ the first j elements of the permutation $X_{k+2i}, \ldots, X_1$ always has more values $+1$ than -1, it follows from the ballot theorem that $P(N_k = k + 2i | S_{k+2i} = k) = k/(k + 2i)$, which proves the result. ■

https://doi.org/10.1017/S0269964820000212 Published online by Cambridge University Press

Letting $g_p(i) = P(N_k = k + 2i)$, it follows from Lemma 2 that

$$\frac{g_p(i+1)}{g_p(i)} = \frac{(k+2i)(k+2i+1)}{(k+i+1)(i+1)} p(1-p), \quad i \geq 0. \tag{3.1}$$

With $G_p(i) \equiv P(N_k \leq k + 2i)$, Eq. (3.1) can be used to recursively compute both $g_p(i)$ and $G_p(i)$.

With $N(t) \equiv N_1(t) + N_2(t)$, it follows that $\{N(t), t \geq 0\}$ is a Poisson process with rate $\lambda + \mu$. Because the event times of the Poisson process $\{N(t), t \geq 0\}$ are independent of whether these events are from the N_1 or N_2 process, it follows that T_k is distributed as the time of the N_k^{th} event of the Poisson process $\{N(t), t \geq 0\}$, where N_k is independent of this Poisson process.

For $p^* = \lambda/(\lambda + \mu)$, let $g(i) = g_{p^*}(i)$, $G(i) = G_{p^*}(i)$.

PROPOSITION 1: *With $v = \lambda + \mu$, $p = \lambda/(\lambda + \mu)$, and with $[x]$ denoting the largest integer less than or equal to x,*

$$P(T_k \leq t) = \sum_{j=0}^{\infty} G([j/2]) e^{-vt} (vt)^{j+k}/(j+k)!$$

PROOF:

$$P(T_k \leq t) = P(N_k \leq N(t)).$$

Conditioning on $N(t)$ and using that $N(t)$ and N_k are independent now yields

$$P(T_k \leq t) = \sum_{j=0}^{\infty} P(N_k \leq k + j) e^{-vt} (vt)^{j+k}/(j+k)!$$

which gives the result. ■

Because $P(T_k \leq t) =$

$$\sum_{j=0}^{m} G([j/2]) e^{-vt} (vt)^{j+k}/(j+k)! + P(N_k \leq N(t) | N(t) > m + k) P(N(t) > m + k),$$

we can approximate $P(T_k \leq t)$ by $\sum_{j=0}^{m} G([j/2]) e^{-vt} (vt)^{j+k}/(j+k)!$ with an approximation error of at most $P(N(t) > m + k)$. Another approach is to use simulation to estimate $P(N_k \leq N(t) | N(t) > m + k)$. An efficient conditional expectation simulation estimator can be obtained by first generating $N(t)$ conditional on it exceeding $m + k$. (An efficient way of doing this is to use the discrete inverse transform algorithm; see Ross [16, Chapter 11]. If the simulated value is $N(t) = r + k$, then return the estimate $P(N_k \leq r + k) = G([r/2])$. Because, with m large, the simulated conditional value of $N(t)$ will, with high probability, be close to $m + k + 1$, the estimator should have a very small variance, and not many simulation runs (perhaps only one) will be needed.

REMARKS 1:

1. *If we let $N_k(\lambda, \mu) = N_k$ and $T_k(\lambda, \mu) = T_k$, then it is well known (see, e.g., p. 211 of [16]) that the probability that the simple random walk with $p = \lambda/(\lambda + \mu)$ is ever*

https://doi.org/10.1017/S0269964820000212 Published online by Cambridge University Press

goes up k is equal to $\min(1, (\lambda/\mu)^k)$. *Consequently,*

$$P(N_k(\lambda,\mu) < \infty) = P(T_k(\lambda,\mu) < \infty) = \begin{cases} 1, & \text{if } \lambda \geq \mu \\ (\lambda/\mu)^k, & \text{if } \lambda < \mu \end{cases}$$

In addition, it is also known that given that the simple random walk with parameter $p < 1/2$ ever goes up k, the sequence of moves up to the time it hits k has the same distribution as the sequence of moves until the simple random walk with parameter $1-p$ hits k. Consequently,

$$P(T_k(\mu,\lambda) \leq t) = (\mu/\lambda)^k P(T_k(\lambda,\mu) \leq t), \quad \mu \leq \lambda$$

2. *Let $E = \sum_{j=0}^{m} G([j/2]) e^{-vt}(vt)^{j+k}/(j+k)!$. If $\lambda < \mu$, Remark 1 gives*

$$E < P(T_k \leq t) < E + (\lambda/\mu)^k P(N(t) > m+k).$$

3. *Whereas Proposition 1 derives $P(T_k \leq t)$ by conditioning on $N(t)$, the formula given in [5] can be obtained by conditioning on N_k. Indeed, doing so yields*

$$\begin{aligned} P(T_k \leq t) &= \sum_{j=0}^{\infty} P(T_k \leq t | N_k = k+2j) P(N_k = k+2j) \\ &= \sum_{j=0}^{\infty} P(S_{k+2j} \leq t) g(j), \end{aligned}$$

here S_{k+2j} is the time of event $k+2j$ of a Poisson process with rate v. However, as noted earlier, this formula is less useful than the one given by Proposition 1.

3.2. Edgeworth Approximation

The conditional waiting time (in discrete time) for a simple random walk to reach 1, given that it does, has pgf

$$g(s) = \frac{1 - \sqrt{1 - 4pqs^2}}{2ps}$$

for $p < 1/2$.

For the continuous time version, take $\lambda + \mu = 1$, $p = \mu < 1/2$, $q = \lambda = 1 - \mu = p$, then,

$$T_k \sim \sum_{i=1}^{N} X_i,$$

where X_i are i.i.d. exponential (1) random variables, satisfies

$$\mathcal{L}(s) = E[e^{-sT_k}] = E\left[\left(\frac{1}{1+s}\right)^N\right] = g\left(\frac{1}{1+s}\right) = \frac{(1+s) - \sqrt{(1+s)^2 - 4pq}}{2p} = \frac{1+s-L(s)}{2p}$$

where

$$L(s) = \sqrt{(1+s)^2 - 4pq}, \quad L'(s) = \frac{1+s}{L(s)}, \quad L(0) = q - p, L^2(0) = 1 - 4pq.$$

https://doi.org/10.1017/S0269964820000212 Published online by Cambridge University Press

It follows that

$$\mathcal{L}'(s) = \frac{L(s) - L(s)'}{2p} = \frac{1 - \frac{1+s}{L}}{2p} = -\frac{\mathcal{L}(s)}{L(s)}.$$

Thus, the cumulant generating function $K(s) = \log(\mathcal{L}(s))$ satisfies the key simplifying relationship, where we use the notation $L \equiv L(s)$,

$$K'(s) = \mathcal{L}'(s)/\mathcal{L}(s) = -1/L$$

and thus

$$E[T_k] = -K'(0) = \frac{1}{q-p}.$$

Next,

$$K''(s) = L^{-2}\left(\frac{1+s}{L}\right) = \frac{1+s}{L^3}$$

so that

$$\mathrm{Var}(T) = K''(0) = \frac{1}{(q-p)^3}.$$

Continuing,

$$K'''(s) = \frac{1}{L^3} - \frac{3(1+s)^2}{L^5}$$

which gives the third central moment

$$E[(T - E[T])^3] = -K'''(0) = \frac{3}{(q-p)^5} - \frac{1}{(q-p)^3} = \frac{3 - (1-4pq)}{(q-p)^5} = \frac{2(1+2pq)}{(q-p)^5}.$$

This gives

$$\gamma \equiv \frac{E(T - E[T])^3}{(\mathrm{Var}(T))^{3/2}} = \frac{6(1+6pq)}{q-p}$$

for the skewness γ of T.

Next,

$$K''''(s) = \frac{15(1+s)^3}{L^7} - \frac{9(1+s)}{L^5},$$

thus

$$E[(T - E[T])^4] - 3(\mathrm{Var}(T))^2 = K''''(0) = \frac{3}{(q-p)^7}[5 - 3(1-4pq)] = \frac{6(1+6pq)}{(q-p)^7}$$

and

$$\tau = \frac{E[(T - E[T])^4] - 3(\mathrm{Var}(T))^2}{(\mathrm{Var}(T))^2} = \frac{6(1+6pq)}{q-p}$$

for the kurtosis τ.

https://doi.org/10.1017/S0269964820000212 Published online by Cambridge University Press

3.3. Illustration of Approximation

We next illustrate the Edgeworth approximation (see [7]).

$$\mathbb{P}\left(\frac{T_k - \mathbb{E}[T_k|T_k<\infty]}{\sqrt{\mathrm{Var}(T_k|T_k<\infty)}} \le t|T_k<\infty\right)$$
$$\approx \Phi(t) - \phi(t)\left[\frac{\gamma(t^2-1)}{6\sqrt{k}} + \frac{1}{72k}[3\tau(t^3-3t) + \gamma^2(t^5-10t^3+15t)]\right],$$

where

$$\gamma = \frac{2(1+2pq)}{\sqrt{q-p}},$$
$$\tau = \frac{6(1+6pq)}{q-p}$$

and Φ, ϕ, respectively are the normal cdf and pdf. We label this the "Edgeworth approximation" in Table 1, and we compare it with the "Normal approximation"

$$\mathbb{P}\left(\frac{T_k - \mathbb{E}[T_k|T_k<\infty]}{\sqrt{\mathrm{Var}(T_k|T_k<\infty)}} \le t|T_k<\infty\right) \approx \Phi(t)$$
$$\mathbb{P}\left(\frac{T_k - \mathbb{E}T_k}{\sqrt{\mathrm{Var}(T_k)}}\right) \approx \Phi(t).$$

The exact probabilities are estimated with simulation. As can be seen, the performance of the Edgeworth approximation is a significant improvement over the normal approximation.

4. SUMMARY AND FUTURE WORK

The reliability of data stored in a cryptocurrency blockchain is an issue that has attracted many researchers. There are many possible avenues for attack by malicious actors, and in this article, we study an ingredient used to analyze the double-spend attack. In particular, we study the distribution of time for one Poisson process to first exceed another by a given number. We derive an exact formula, show how it can be quickly simulated, and show that an Edgeworth approximation performs better than the normal approximation. With many cryptocurrencies, the accepted list of confirmed transactions grows at approximately the times of a Poisson process created by the honest participants, and a double-spend attacker can reverse transactions if the another slower Poisson process they control can exceed the counts of the process created by the honest participants by a given number. The results of this paper provide a method for estimating the distribution of the duration of this type of attack.

Possible avenues for future work include studying the case where the first Poisson process is not homogeneous but has a linearly increasing rate; this covers the more realistic setting discussed in [2] where the computational power of the honest participants is increasing over time. Another avenue for future work is to study the setting where the Poisson process is generalized to a renewal process, possibly with gamma inter-event times; this problem arises in other transaction confirmation methods, for example in [1].

https://doi.org/10.1017/S0269964820000212 Published online by Cambridge University Press

TABLE 1. Comparing approximations for $p = 0.3 = 1 - q$

t	$\mathbb{P}(T_{10} \leq t)$ (simulated)	Normal approximation	Edgeworth approximation
10	0.00098	0.03622401	−0.013199033
11	0.00211	0.04172831	−0.009978425
12	0.00426	0.04789127	−0.005439885
13	0.00747	0.05476233	0.000566464
14	0.01215	0.06239025	0.008177617
15	0.01846	0.07082235	0.017513017
16	0.02678	0.08010374	0.028670129
17	0.03749	0.09027648	0.041720469
18	0.05102	0.10137873	0.056706258
19	0.06608	0.11344385	0.073637894
20	0.08429	0.12649953	0.092492359
21	0.10456	0.14056698	0.113212685
22	0.12766	0.15566009	0.135708530
23	0.15275	0.17178473	0.159857871
24	0.17835	0.18893808	0.185509782
25	0.20632	0.20710809	0.212488184
26	0.23569	0.22627311	0.240596445
27	0.26606	0.24640160	0.269622618
28	0.29584	0.26745202	0.299345119
29	0.32666	0.28937289	0.329538592
30	0.35795	0.31210306	0.359979712
31	0.38926	0.33557209	0.390452686
32	0.41956	0.35970086	0.420754218
33	0.44892	0.38440235	0.450697755
34	0.47840	0.40958251	0.480116844
35	0.50662	0.43514139	0.508867514
36	0.53498	0.46097425	0.536829612
37	0.56155	0.48697292	0.563907126
38	0.58718	0.51302708	0.590027531
39	0.61224	0.53902575	0.615140276
40	0.63631	0.56485861	0.639214574
41	0.65934	0.59041749	0.662236646
42	0.68073	0.61559765	0.684206640
43	0.70145	0.64029914	0.705135416
44	0.72153	0.66442791	0.725041391
45	0.74035	0.68789694	0.743947620
46	0.75824	0.71062711	0.761879252
47	0.77483	0.73254798	0.778861480
48	0.79004	0.75359840	0.794918048
49	0.80493	0.77372689	0.810070342
50	0.81923	0.79289191	0.824337044

Acknowledgments

We would like to thank the referees and the editor for their wise and helpful comments.

FUNDING

The research of Sheldon Ross is based on work supported by, or in part, by the National Science Foundation under contract/grant CMMI1662442.

https://doi.org/10.1017/S0269964820000212 Published online by Cambridge University Press

References

1. Bissian, G. & Levine, B. (2017). Bobtail: A proof-of-work target that minimizes blockchain mining variance. Accepted to appear in *ISOC NDSS 2020*.
2. Bowden, R., Keeler, P., Krzesinski, A.E., & Taylor, P.G. (2018). Block arrivals in the Bitcoin blockchain. https://arxiv.org/pdf/1801.07447.pdf.
3. Eyal, I. & Sirer, E.G. (2018). Majority is not enough: Bitcoin mining is vulnerable. *Communications of the ACM* 61(7): 95–102.
4. Georgiadis, E. & Zeilberger, D. (2019). A combinatorial-probabilistic analysis of bitcoin attacks. *Journal of Difference Equations and Applications* 25(1): 56–63.
5. Gofffard, P.-O. (2019). Fraud risk assessment within blockchain transactions. *Advances in Applied Probability* 51(2): 443–467.
6. Grunspan, C. & Perez-Marco, R. (2018). Double spend races. *International Journal of Theoretical and Applied Finance* 21(8): 1850053.
7. Hall, P. (1992). Principles of edgeworth expansion. In *The bootstrap and edgeworth expansion*. Springer Series in Statistics. New York, NY: Springer.
8. Lin, I.-C. & Liao, T.-C. (2017). A survey of blockchain security issues and challenges. *IJ Network Security* 19(5): 653–659.
9. Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf.
10. Narayanan, A., Bonneau, J., Felten, E., Miller, A., & Goldfeder, S (2016). *Bitcoin and cryptocurrency technologies: A comprehensive introduction*. Princeton, NJ: Princeton University Press.
11. Orcutt, M. (2019). Once hailed as unhackable, blockchains are now getting hacked. *MIT Technology Review*, February 19, 2019.
12. Ozisik, A.P., Andresen, G., Bissias, G., Houmansadr, A., & Levine, B.N. (2016). A secure, efficient, and transparent network architecture for bitcoin. UMass Amherst, Tech. Rep. UM-CS-2016.
13. Ozisik, A.P. & Levine, B.N. (2017). An explanation of Nakamoto's analysis of double-spend attacks. arXiv preprint arXiv:1701.03977.
14. Pinzon, C. & Rocha, C. (2016). Double-spend attack models with time advantange for bitcoin. *Electronic Notes in Theoretical Computer Science* 329: 79–103.
15. Rosenfeld, M. (2012). Analysis of hashrate-based doublespending. https://www.bitcoil.co.il/Doublespend.pdf.
16. Ross, S. (2019). *Introduction to probability models*, 12th ed. Academic Press.
17. Saia, R., Carta, S., Reforgiato Recupero, D., & Fenu, G. (2019). Internet of entities (IoE): A blockchain-based distributed paradigm for data exchange between wireless-based devices. In *Conference: 8th International Conference on Sensor Networks (SENSORNETS-2019)*, Prague, Czech Republic.
18. Wang, H., Zheng, Z., Xie, S., Dai, H.-N., & Chen, X. (2018). Blockchain challenges and opportunities: A survey. *International Journal of Web and Grid Services* 14(4): 352.

https://doi.org/10.1017/S0269964820000212 Published online by Cambridge University Press