mathematics

MDPI

Article

# Achieving Security in Proof-of-Proof Protocol with Non-Zero Synchronization Time

**Lyudmila Kovalchuk [1,*], Volodymyr Kostanda [2], Oleksandr Marukhnenko [3] and Oleksii Pozhylenkov [4,*]**

1 Physical Technical Institute, NTUU "Igor Sikorsky Kyiv Polytechnical Institute", 03056 Kiev, Ukraine
2 Adoriasoft, 61166 Kharkiv, Ukraine; ceo@adoriasoft.com
3 Faculty of Computer Engineering and Control, Kharkiv National University of Radioelectronics, 61166 Kharkiv, Ukraine; oleksandr_marukhnenko@adoriasoft.com
4 Department of Mathematics, Physics and IT, Odesa Mechnikov University, 65000 Odesa, Ukraine
* Correspondence: lusi.kovalchuk@gmail.com (L.K.); alex_pozhilenkov@adoriasoft.com (O.P.)

**Abstract:** Among the most significant problems that almost any blockchain faces are the problems of increasing its throughput (i.e., the number of transactions per unit of time) and the problem of a long waiting time before block confirmation. Thus, for example, in the most common BTC blockchain, according to various estimates, throughput is from 3 to 7 tps (transactions per second), and the average block confirmation time (block is considered confirmed if it has at least 6 blocks over it) is 1 h. At the same time, it is impossible to solve these problems directly by increasing the block size or increasing block generation intensity because this leads to essentially a decrease in the security of the blockchain in the first turn against double spend and splitting attacks. Such problems lead to the inconvenience of the practical use of cryptocurrencies to pay for goods and services. Proposed a few years ago, the PoP consensus protocol potentially helps to solve the problem of increasing blockchain throughput, although it was originally intended to ensure the stability of "young" blockchains, with "small" PoW, through the use of a secure blockchain, such as BTC. A blockchain that has provable security is called the security-provided blockchain (SPB), and one that uses SPB to achieve its security is called the security-inherited blockchain. In this paper, we give explicit formulas which describe how the number of confirmation blocks in the security-inherited blockchain, which is sufficient to achieve a given security level of this blockchain to a double spend attack, depends on the parameters of both blockchains. It is essential that we use a realistic model to obtain the results, taking into account the synchronization times of both blockchains. Such a model is much closer to the real situation, but at the same time, it leads to significant analytical difficulties in obtaining results. The obtained formulas are convenient for numerical calculations, the numerous examples of which are also given in this work.

**Keywords:** PoP consensus; non-zero synchronization; blockchain security

**MSC:** 68M01; 60G40; 91A60; 33B20

check for updates

**Citation:** Kovalchuk, L.; Kostanda, V.; Marukhnenko, O.; Pozhylenkov, O. Achieving Security in Proof-of-Proof Protocol with Non-Zero Synchronization Time. *Mathematics* **2022**, *10*, 2422. https://doi.org/10.3390/math10142422

Academic Editors: Marina Alexandra Pedro Andrade and Maria Alves Teodoro

Received: 30 May 2022
Accepted: 8 July 2022
Published: 11 July 2022



## 1. Introduction

This paper aims to analyze the security of the Proof-of-Proof (PoP) protocol described in [1]. The primary purpose of this protocol is to strengthen any "light" blockchain–blockchain, where blocks are created with small Proof-of-Work (PoW), but it may also be applied to blockchains with arbitrary consensus protocols to increase their security by binding to a blockchain, which is provably secure.

Note that block intensity generation may be relatively high for blockchains with "small" PoW, so transactions are processed quicker than in a classical "heavy" and secure blockchain. However, it is known that the security of blockchain (for example, against double-spend attacks) decreases when block intensity generation increases. According to various estimates, the number of transactions per second is expected to range from 3 to

7 for the secure BTC blockchain, with the desired speed of up to several thousand. We can say that such "light" blockchains increase throughput and lose out in security. This is the case when the PoP protocol can be used to provide a sufficient level of security for such "light" blockchains with high throughput (or even for blockDAG, but after some significant modification). In other words, it helps to solve one of the main problems of blockchains—the low network throughput.

In the available literature, we can find a plethora of different suggestions for increasing the throughput. These suggestions can be roughly divided into two types:

1. Transition from blockchain to block graph (DAG—directed acyclic graph);
2. Use various "add-ons" over the blockchain.

The first type includes papers [2–6]. The authors of these papers announce improvements in both throughput and latency until the transaction is fully confirmed. However, none of these papers contains rigorous proofs of the stated results, semiempirical explanations are, at best, what they provide. Some of them also have serious mathematical errors (for example, refs. [2–5]).

Papers of the second type [7–10] contain more substantiated statements and more rigorous proofs. However, they solve only one of the existing problems—increasing the throughput. The second currently remains unresolved.

The main idea of the second type of papers is that, in addition to the classic "secure and stable" blockchain (with "slow" block generation, with liveness and consistence properties), which is called MainChain (MC), one can generate additional "separate" blocks or even blockchains that can be produced as quickly as you like (with minimal PoW or Proof-of-Stake (PoS)). Here, we will call such blockchains SideChains (SCs), though they may also have other names (parachain, fruitchain, and so on). SC blocks can refer to each other and to the MainChain. MC blocks (or some of them) may refer directly to SC blocks or may contain some information about a recent SC block, which can be considered as some type of reference.

The SC block $B$ is considered stable (i.e., such that its transactions are irreversible with a probability close to 1) if it is referenced by any stable block of the MC blockchain. That is, the stabilization of a SC block is still a consequence of stabilizing the corresponding block from the MC. Therefore, the time until the block stabilizes remains long.

The PoP protocol suggested in [1] can also be classified as a type 2 blockchain, but its main difference from the protocols indicated in [7–11] is that it uses a "foreign" blockchain as an MC (for instance, BTC blockchain). In this case, there should be a "two-way" communication with the blockchains (i.e., references).

The Veriblock (VB) blockchain with PoP consensus refers to blocks from the MC, and the MC must, at certain limited intervals, refer to the VB. The blockchain that is the MC is called a security-provided blockchain (SP blockchain) and the VB blockchain is called a security-inherited blockchain (SI blockchain). This protocol increases the throughput (i.e., tps) significantly, but the transaction confirmation time is still fully determined by the "slow" SP blockchain. In addition to this, the VB blockchain itself can act as SP-blockchain for other, different blockchains.

In what follows, we give a short description of the main features of the PoP protocol and then analyze in more detail the procedure of achieving security in this protocol.

We also emphasize that it is essential to consider block synchronization time or time delay for block sharing in security analysis. The importance of this parameter was shown in many works, mainly [12–15]. Notably, it was shown that if the adversary is well synchronized and honest miners are not, the security threshold (the minimal ratio of adversary, which can attack the blockchain with probability 1 despite the number of confirmation blocks) decreases dramatically. For example, as it was shown in [14], for Bitcoin, if the block synchronization time is 20 s and $\alpha = \frac{1}{600}$, the security threshold is about 49%; if the block synchronization time is 60 s, the security threshold is about 47.5%. It means that if the ratio of an adversary is not less than 0.49 or 0.475, respectively, his attack will be successful with probability 1 and it cannot be prevented with a large number of

confirmation blocks. When the adversary's ratio is smaller than this security threshold, we need to increase the number of confirmation blocks (in comparison with the case of zero synchronization time) to achieve the same security level.

Our security analysis in this work assumes that the synchronization time is non-zero and upperbound with some known value. For example, for the BTC blockchain, the most widely used assumption is that the synchronization time is up to 20 s. We obtain additional confirmation of the non-zero synchronization time in BTC from its regular "forks", which occurs about 6 times in a month, on average. Thus, such forks (without a doublespend attack) occur when two different mining pools create blocks (of the same height) within a time delay interval.

Of course, we cannot state that time delay is essentially large all the time, but since we cannot define or predict the period when it really is essentially large, it is better to assume that the synchronization time is non-zero all the time in order to guarantee security in the worst case.

The main contribution of this work is the proving of explicit expressions, obtained for model with non-zero time delay, for the calculation of the number of confirmation blocks, which is enough for block stabilization in the SI blockchain. In particular, our results give the following possibilities:

1. To calculate the probability of double spend attack for given network parameters (block generation intensity, time delay for block sharing) and given number of blocks generated after the block with transaction (the number of so-called confirmation blocks);
2. Given the network parameters and preset (small) probability, to calculate the number of confirmation blocks which guarantees that the probability of double spend attack is not larger than the preset value.

This results are completely new to the realistic model with non-zero time delay. Some known results for synchronous model may be also derived from setting the time delay equal to zero.

## 2. Materials and Methods

In this section, we first explain the basic idea behind the PoP protocol, as well as some of the details and specifications of this protocol given in [1]. Next, some of the auxiliary statements that are necessary for obtaining the key result are proved. The main result is Theorem 1, which gives the possibility to calculate the required number of confirmation blocks in the SI blockchain, guaranteeing (with overwhelming probability) the stability of a certain block in it.

### *2.1. PoP Consensus Protocol*

The PoP protocol uses the properties of the SP blockchain (liveness, consistence) to provide similar properties to the SI blockchain. Hereinafter, we will assume that the SP blockchain is BTC, and the SI blockchain is VB.

In the SI blockchain, each subsequent block refers to several previous ones according to a specific rule, depending on the parameters of the network.

We introduce the concept of keystone block in order to set the link rule. In addition to this, two parameters related to the SI blockchain and one parameter related to the SP blockchain are set.

The keystone block is every $i$th block, where $i \geq 2$ is the so-called keystone interval. Each new block in the SI blockchain refers to the previous block, and the $r$ of the last keystone blocks, where $r \geq 2$ is the number referenced keystones.

Therefore, each block contains a reference to $r$ or $r + 1$ of keystone blocks ($r + 1$—if the block comes immediately after the keystone block).

The VB paper says that each block always refers to precisely $i$ of keystone blocks (table on page 22). However, the second table on page 23 provides an example that

contradicts the statement on page 22 but agrees with our statement (that refers to $r$ or $r+1$ of keystone blocks).

We want to emphasize that two parameters $i$ and $r$ relate only to the SI blockchain.

The third parameter, $d$, relates to the SP blockchain. It shows that, to maintain the validity of the SI blockchain, the interval between references to new keystone blocks does not exceed $d$ of blocks in the SP blockchain, i.e., if at some point, a block number $l$ in the SP blockchain refers to a keystone block number $k$ in the SI blockchain, then no later than in a block number $l+d$ in the SP blockchain, a link to $l+1$th keystone block should appear.

### 2.2. Achieving Stability in the SI Blockchain

Informally speaking, the idea to achieve stability in the SI blockchain using stability in the SP blockchain can be described as follows: block $B$ in the SI blockchain is stable if the block $B^*$ in the SP blockchain is stable, where $B^*$ is the first block in the SP blockchain that refers to the block $B$. In other words, in order to "cancel" block $B$ in the SI blockchain, one needs to perform a long enough fork not only in the SI blockchain, but also in the SP blockchain, which is an arduous computational task. To provide some numerical characteristics to "stability", the concepts of $N$-BTC-References and N-BTC-Finality are introduced. The first of them, $N$-BTC-References, means that after block $B^*$ in the SP blockchain, $N$ blocks have already been created, and if an attacker begins to build an alternative branch in the SI blockchain, in which block $B$ is absent, then a necessary condition for its validity is the appearance of reference to its blocks in the SP blockchain. In the paper, this is called an "early attack detection metric". That is, if after block $B^*$ in the SP blockchain $N$ blocks have already been released, and there are no references to the alternative chain in them, then this alternative chain does not exist (someone might have started building it, but now it has lost its validity).

The $N$-BTC-Finality term means that $N$ blocks after block $B^*$ have already been created in the SP blockchain, and this amount is sufficient to guarantee the stability of block $B^*$ with a probability almost indistinguishable from 1. Please note that this probability depends on three parameters: the intensity of block generation (for BTC, it is $\frac{1}{600}$), the ratio of the attacker's hash rate, and the network synchronization time (that is, the block propagation delay time).

That is, the main idea of achieving stability in the SI blockchain can be described as follows:

1. We wait until block $B^*$ appears in the SP blockchain with reference to block $B$ from the SI blockchain;
2. After this moment, we wait for the creation of $N$ blocks after block $B^*$, where the value $N$ is determined by the above parameters of the SP blockchain (block generation intensity, the ratio of the attacker's hash rate, network synchronization time), as well as the desired probability value, which we choose ourselves.

At the same time, the $N$-BTC-References parameter is intermediate. It simply allows to detect an attack at an early stage but does not guarantee the impossibility of a fork in the SP blockchain with the required (set by us) probability.

The above idea of achieving stability in a "light" blockchain using some other "heavy and stable" blockchain is proposed (by the authors of [1]) to be generalized and developed further, with the possibility of using not only for the VeriBlock blockchain, but also for any existing altcoin. In this case, the VeriBlock blockchain already acts as an SP blockchain, and the altcoin blockchain, respectively, is an SI blockchain. In this case, to characterize the degree of stability achievement, by analogy, the following parameters are used:

1. $N$-BTC-References;
2. $N$-BTC-Finality;
3. $N$-VBK-References;
4. $N$-VBK-Finality.

### 2.3. Correlation of the Number of Blocks Created in Different (Independent) Blockchain

A measure of block stability in the blockchain is usually the number of blocks created after it, provided there is no visible fork in this interval. For a splitting attack, at the moment, no working (non-asymptotic) estimates of the attack probability have been obtained, depending on the block depth, for a model with continuous time. For a double spend attack, such estimates were obtained for various mathematical models (with continuous time [12,14–16] and with discrete [15]; without taking into account the block synchronization time [16] and taking it into account [12–15]). Although it is not completely clear from [1], it can be reasonably assumed that the authors used the same quantitative characteristic to guarantee the stability of a block in the SI blockchain. Informally speaking, the reasoning is roughly as follows.

Let us assume that block $B^*$ is the first block in the SP blockchain that refers to a block $B$ from the SI blockchain. Then, provided that there are no references to some alternative branch of the SI blockchain in the blocks following the block $B^*$, to build this alternative branch of the SI blockchain, you need to fork the SP blockchain, that is, build its alternative branch in which there is no block $B^*$. The greater the depth block $B^*$ is, the smaller the probability is to create such an alternative branch that will be longer than the existing one. Moreover, we ourselves can set the value of such a probability and, accordingly, choose the block depth in the SP blockchain, which guarantees that the probability of a fork will be no more than a given value. For example, for the probability of a fork to be no more than a certain small $\varepsilon$, we just need to build $n$ blocks after the block $B^*$. After these blocks are created, the probability of removing the $B$ block from the SI blockchain also does not exceed $\varepsilon$. We do not want to look into the SP blockchain every time to check how many blocks there are, and we want to estimate the probability of a fork only by the number of blocks generated in the SI blockchain. Then, we set a small $\delta$ and determine value $k$ such that the probability of the event $A_{n,\geq k} = \{$not less than the $kSP$ blocks that occur during the time when $n$ SI blocks occur$\}$ is not less than $1 - \delta$. With the obtained value $k$, the probability of removing block $B$ from the SI blockchain will not be larger than $\varepsilon + \delta$. That is, initially, we set the desired upper bound $\gamma$ on the fork probability and then we determine values $n$ and $k$ in a such way that the corresponding sum $\varepsilon + \delta$ is not larger than $\gamma$.

The paper provides formulas that are auxiliary for calculating the probability of $A_{n,\geq k}$ event (page 11, 12 , and on). These formulas correspond to the probabilities of exponential and inverse binomial distributions. They are correct if the following conditions are met with regard to the considered SP blockchain and SI blockchain:

1. The synchronization time in both blockchains is zero, that is, after creating a block, all nodes instantly see it;
2. Both blockchains use the PoW consensus protocol.

If at least one of these conditions is violated, then the basic formulae for calculating the probability of creation of exactly $k$ blocks in one blockchain during the time until $n$ blocks are created in another blockchain will be completely different.

In our analysis given below, we get rid of the first assumption and build correspondent probability estimations for the blockchains with non-zero synchronization time bounded with some arbitrary value.

### 2.4. Main Assumptions, Designations, and Some Auxiliary Statements

In this section, we describe the basic assumptions of our model and introduce the main designations. Sometimes, we will also refer to some statements proved in [16].

We will use SP blockchain for "Security provided blockchain" and SI blockchain for "Security inherited blockchain". Let us define the following random variables (RVs):

$T_P$—the RV that is measuring the time it takes to mine a block in the SP blockchain,

$T'_P$—the RV that is measuring the time it takes to mine and share the block in the SP blockchain,

$T_I$—the RV that is measuring the time it takes to mine a block in the SI blockchain,

$T'_I$—the RV that is measuring the time it takes to mine and share the block in the SI blockchain.

As shown in [16], RVs $T_I$ and $T_P$ have exponential distributions:

$$\begin{aligned} F_{T_P}(t) &= P(T_P < t) = 1 - e^{-\alpha_P t}, \\ F_{T_I}(t) &= P(T_I < t) = 1 - e^{-\alpha_I t}, \end{aligned} \tag{1}$$

for some $\alpha_P > 0, \alpha_I > 0$. The physical sense of these two parameters is that $\frac{1}{\alpha_P}$ and $\frac{1}{\alpha_I}$ are the mean rates of block generation in the SP blockchain and SI blockchain, correspondingly. Define $\alpha = \alpha_P + \alpha_I$.

We also assume that $D_P$ denotes the time it takes in the SP blockchain to share a block (after it was generated) for all nodes in the network. The value $D_I$ is the corresponding time in the SI blockchain. In this very work, we assume that $D_I = 0$. The reasons for this assumption are as follows:

1. From a security consideration, it is critical that we can guarantee (with a probability close to 1) that during the time of the generation of a certain number of blocks in the SI blockchain, at least a certain number of blocks in the SP blockchain are created;
2. The most "terrible" thing that can happen from this assumption is that we will spend "extra" time waiting for a slightly larger number of confirmation blocks in the SI blockchain than is necessary to achieve the declared security, as a result of which, the obtained security will be somewhat higher than the declared one;
3. Under the assumption that in both blockchains the synchronization time is nonzero, the mathematical model becomes much more complicated, making it almost impossible to obtain the results we need.

Our assumption means that $T'_I = T_I$ and $F_{T'_I}(x) = F_{T_I}(x)$.

We also should note that, for the sake of simplicity, we assume that the block delay time $D_P$ is the same for all blocks in the SP blockchain. Of course, this is a kind of simplification of the real model, but in an alternative case, it is impossible to take into account all particular time delays. On the other hand, we can consider $D_P$ the largest time delay in the SP blockchain and consider setting the problem "in the worst-case scenario", that is, when all blocks in the SP blockchain are delivered with the maximum delay. As with the previous assumptions, this also leads to an increase in the blockchain security compared to the declared one.

In these designations, we have

$$T'_P = D_P + T_P, T'_I = T_I. \tag{2}$$

Define $p_P$ the probability that the next block in the SP blockchain will be generated before the next block in the SI blockchain (i.e., faster than in the SI blockchain), and $p_I = 1 - p_P -$ is the probability of the opposite event. Using considerations very similar to those in [16], we obtain

$$p_P = \frac{\alpha_P}{\alpha_P + \alpha_I}, \quad p_I = \frac{\alpha_I}{\alpha_P + \alpha_I}. \tag{3}$$

Actually, in our case, we are interested in two other values that take into account the time delay $D_P > 0$. We introduce these values as follows:

$p'_P$—the probability that the next block in the SP blockchain will be generated and shared before the next block in the SI blockchain will be generated and shared for all nodes;

$p'_I$—the probability of the alternative event, $p'_I = 1 - p'_P$.

According to their definitions,

$$p'_P = P(T'_P < T_I), p'_I = P(T_I \leq T'_P), \tag{4}$$

and also $p'_P + p'_I = 1$.

These two values in (4) are much more important than the values in (3) because they take into account time delay $D_P$ and describe the state of the network much more realistically. In what follows, we will show that the relation between the number of blocks in the SP blockchain and the SI blockchain depends on these very values in (4) rather than values in (3). Thus, if $D_P$ is rather large, the "real" hash rate $p'_P$ in the SP blockchain is essentially smaller than $p_P$.

Now we are going to find $p'_P$ and $p'_I$.

**Lemma 1.** *In our designations, the next equalities hold:*

$$\begin{aligned} p'_I &= 1 - e^{-\alpha_I D_P} \times \frac{\alpha_P}{\alpha_I + \alpha_P} = 1 - e^{-\alpha_I D_P} \times p_P; \\ p'_P &= e^{-\alpha_I D_P} \times \frac{\alpha_P}{\alpha_I + \alpha_P} = e^{-\alpha_I D_P} \times p_P. \end{aligned} \tag{5}$$

**Proof.** Note that distribution functions for RVs $T'_P$ and $T_B$ according to (2) and (4) are

$$\begin{aligned} F_{T'_P}(t) &= P\big(T'_P < t\big) = P(T_P + D_P < t) = \\ &= P(T_P < t - D_P) = \begin{cases} 1 - e^{-\alpha_P(t - D_P)}, \text{ if } t > D_P \\ 0, \text{ else;} \end{cases} \end{aligned} \tag{6}$$

$$F_{T_I}(t) = 1 - e^{-\alpha_I t}. \tag{7}$$

The corresponding densities are

$$\begin{aligned} f_{T'_P}(t) &= \alpha_P e^{-\alpha_P(t - D_P)}; \\ f_{T_I}(t) &= \alpha_I e^{-\alpha_I t}. \end{aligned} \tag{8}$$

Then, according to the composite probability formula,

$$\begin{aligned} p'_I &= P\big(T_I < T'_P\big) = \\ &= P\big(T_I < T'_P / T_I < D\big) P(T_I < D) + \\ &+ P\big(T_I < T'_P / T_I > D\big) P(T_I > D). \end{aligned} \tag{9}$$

However, $P(T_I < T'_P / T_I < D_P) = 1$, because from (2) we obtain $T'_P \geq D_P$. So

$$P'_I = P(T_I < D_P) + P\big(D_P < T_I < T'_P\big)$$

Next, according to (4) and (6)–(8),

$$P(T_I < D_P) = 1 - e^{-\alpha_I D_P};$$

$$\begin{aligned} &P\big(D_P < T_I < T'_P\big) = \int_{x,y: D_P < x < y} f_{T_I}(x) f_{T_P}(y) dx dy = \\ &\int_{D_P}^{\infty} \left( \int_{D_P}^{y} f_{T_I}(x) dx \right) f_{T_P}(y) dy = \int_{D_P}^{\infty} \big( F_{T_I}(y) - F_I(D_P) \big) f_{T_P}(y - D_P) dy = \\ &\int_{D_P}^{\infty} \Big( 1 - e^{-\alpha_I y} - \Big( 1 - e^{-\alpha_I D_P} \Big) \Big) \alpha_P e^{-\alpha_P} (y - D_P) dy = \\ &\int_{D_P}^{\infty} \Big( e^{-\alpha_I D_P} - e^{-\alpha_I y} \Big) \alpha_P e^{-\alpha_P} (y - D_P) dy = \\ &\alpha_P e^{-\alpha_I D_P} \int_{D_P}^{\infty} \big( 1 - e^{-\alpha_I} (y - D_P) \big) e^{-\alpha_P (y - D_P)} dy = \\ &\alpha_P e^{-\alpha_I D_P} \int_{0}^{\infty} \big( 1 - e^{-\alpha_I z} \big) e^{-\alpha_P z} dz, \end{aligned}$$

where $z = y - D_P$.

After integration, we obtain

$$P\left(D_P < T_I < T'_P\right) = e^{-\alpha_I D_P} \times \frac{\alpha_I}{\alpha_P + \alpha_I},$$

and from (9), we obtain

$$p'_I = 1 - e^{-\alpha_I D_P} + e^{-\alpha_I D_P} \times \frac{\alpha_I}{\alpha_P + \alpha_I} =$$
$$= 1 - e^{-\alpha_M D_H} \times \left(1 - \frac{\alpha_M}{\alpha_H + \alpha_M}\right) = 1 - e^{-\alpha_M D_H} \times \frac{\alpha_H}{\alpha_H + \alpha_M} =$$
$$= 1 - e^{-\alpha_M D_H} \times \frac{\alpha_H}{\alpha_H + \alpha_M} = 1 - e^{-\alpha_M D_H} \times p_H.$$

Respectively, $p'_P = 1 - p'_I = e^{-\alpha_I D_P} p_P$, and the formulas (5) and the lemma are proved. □

*2.5. Main Results*

In this section, we formulate our main results after some auxiliary lemmas.

Denote $T'_P(i)$ as the time needed in the SP blockchain to form and share the $i$-th block, i.e., the time from the event " $i-1$-th block is formed and available for all nodes" till the event " $i$-th block is formed and available for all nodes". Similar to (2), we can also say that

$$T'_P(i) = T_P(i) + D_P, \tag{10}$$

where $T_P(i)$ is the time needed in the SP blockchain to generate the $i$-th block, after the $i-1$-th block becomes available. Then, $T'_P(i), i \geq 1$, are independent, identically distributed RVs with distribution functions

$$F_{T'_P(i)}(t) = F_{T'_P}(t) = F_{T_P}(t - D_P) = 1 - e^{\alpha_P(t - D_P)}, \quad foralli \geq 1,$$

where the last equality follows from (1).

Additionally, define RVs $T_I(i), i \geq 1$, in the same way. Then their distribution functions are

$$F_{T_I(i)}(t) = 1 - e^{-\alpha_I t}, \text{for all } i \geq 1.$$

In addition, for $n \geq 1$, let us define RVs $S_P(n)$, where

$$S_P(n) = \sum_{i=1}^{n} T_P(i) \tag{11}$$

and RVs $S'_P(n)$, where

$$S'_P(n) = \sum_{i=1}^{n} T'_P(i). \tag{12}$$

Then $S_P(n)$ is the time needed to generate (without sharing) $n$ (independent) blocks in the SP blockchain and $S'_P(n)$ is the time needed to generate and share $n$ blocks in the SP blockchain, one after another.

From (10), we obtain that

$$S'_P = S_P(n) + nD_P,$$

where $S_P(n)$ has an Erlang distribution as the sum of independent identically distributed RVs with exponential distribution:

$$F_{S_P(n)}(t) = P(S_P(n) \leq t) = 1 - e^{-\alpha_P t} \sum_{i=1}^{n} \frac{(\alpha_P t)^k}{k!}. \tag{13}$$

Additionally, define RVs $S_I(n)$ in the same way:

$$S_I(n) = \sum_{i=1}^{n} T_I(i). \tag{14}$$

Note that $S_I(n)$ also has an Erlang distribution:

$$F_{S_I}(n)(t) = 1 - e^{-\alpha_I t} \sum_{i=0}^{n-1} \frac{(\alpha_I t)^k}{k!}. \tag{15}$$

Let us also define the random process (RP) $N_P(t)$ as the number of blocks generated in the SP blockchain during the time interval of the length $t$, if the time delay was equal to zero.

**Lemma 2.** *The RP $N_P(t)$ has Poisson distribution with parameter $\alpha_P t$:*

$$P(N_P(t) = n) = \frac{(\alpha_P t)^n e^{-\alpha_P t}}{n!}. \tag{16}$$

**Proof.** The event $\{N_P(t) = n\}$ is the same as the event $\{S_P(n) < t\} \cap \{S_P(n+1) > t\}$, where $S_P(n)$ was defined in (11). We can write the subsequent chain of equalities:

$$\begin{aligned} \{N_P(t) = n\} &= \{S_P(n) < t \cap S_P(n+1) > t\} = \\ &= \left\{S_P(n) < t \cap \overline{S_P(n+1) < t}\right\} = \\ &= \{S_P(n) < t\} \backslash \{S_P(n+1) < t\}. \end{aligned}$$

Yet, according to the definition (11), $\{S_P(n+1) < t\} \subset \{S_P(n) < t\}$, then, using (15),

$$\begin{aligned} P\{N_P(t) = n\} &= P\{S_P(n) < t\} - P\{S_P(n+1) < t\} = \\ &= F_{S_P(n)}(t) - F_{s_P(n+1)}(t) = \frac{(\alpha_P t)^n e^{-\alpha_P t}}{n!}. \end{aligned}$$

The lemma is proved. □

Note that for RV $N_I(t)$, defined in the same way, we also have the same statement:

$$P(N_I(t) = n) = \frac{(\alpha_I t)^n e^{-\alpha_I t}}{n!} \tag{17}$$

**Notation 1.** *From the properties of the Poison process (independent increments, absence of aftereffects), we get that for any $t_1, t_2 > 0$ the distribution law of $N_M(t_2)$ is the same as the distribution law of*

$$N_M(t_2 + t_1) - N_M(t_1),$$

*i.e., the number of events happening during $[t_1, t_1 + t_2]$ has the same distribution law as the number of events happening during $[0, t_2]$.*

*Additionally, note that the number of events happening during the period $[0, t_1 + t_2]$ is the sum of the numbers of events happening during $[0, t_1]$ and $[t_1, t_1 + t_2]$.*

*We will use this property in the lemma below.*

Additionally, introduce RP $N'_P(t)$ as the number of blocks generated and shared (one after another) in the SP blockchain during the time interval of the length $t$. Then

$$P\big(N'_P(t) = k\big) = P\big(N_P\big(t - kD_p\big) = k\big) = \begin{cases} 0, \text{ if } t \le kD_P, \\ e^{-\alpha_P\left(t - kD_p\right)} \times \frac{\left(\alpha_P\left(t - kD_p\right)\right)^k}{k!}, \text{ else.} \end{cases} \tag{18}$$

Now we can define our purpose as to find, or at least to estimate, the probability

$$P\big(N'_P(S_I(n)) \ge k\big), \tag{19}$$

which is the probability of the following event $A_{n,\ge k}$ : $A_{n,\ge k}$ = "Not less than $k$ blocks were generated and shared in the SP blockchain during the time when exactly $n$ blocks were generated in the SI blockchain (i.e., during the time $S_I(n)$)".

According to our purpose, we need to build lower estimation for $P(A_{n,\geq k})$ and then, for fixed $k$ (which corresponds to definite security level $1-\varepsilon$ in the SP blockchain) and fixed small $\delta > 0$ to define

$$n_0(\delta, k) = \min\{n : P(A_{n,\geq k}) \geq 1-\varepsilon\}.$$

We can also define $n_0(\delta, k)$ as

$$n_0(\delta, k) = \min\{n : P(\neg A_{n,\geq k}) < \varepsilon\}, \tag{20}$$

where $P(\neg A_{n,\geq k}) = P(N'_P(S_I(n)) < k)$, according to (20), (21).

Define

$$C_{n,<k} = \neg A_{n,\geq k} \text{ and } B_{n,t} = \{S_I(n) > t\} \text{ for } n \in \mathrm{N}, t > 0 \tag{21}$$

where

$$P(B_{n,t}) = e^{-\alpha_I t} \times \sum_{l=0}^{n-1} \frac{(\alpha_I t)^l}{l!}. \tag{22}$$

Now, we are ready to formulate our main result about the number of generated blocks.

**Theorem 1.** *In our designations,*

$$\begin{aligned} P(C_{n,<k}) < &\left(\sum_{l=1}^{n-1}\left(\sum_{i=0}^{k-1} C^i_{n-l+i-1} p_P^i p_I^{n-l}\right) \times \frac{e^{-\alpha_l k D_P} \times (\alpha_I k D_P)^{l-1}}{(l-1)!} + \frac{e^{-\alpha_l k D_P} \times (\alpha_I k D_P)^{n-1}}{(n-1)!}\right) \times \\ &\times e^{-\alpha_t k D_P} \times \sum_{l=0}^{n-1} \frac{(\alpha_I k D_P)^l}{l!} + \left(1 - e^{-\alpha_l k D_P} \times \sum_{l=0}^{n-1} \frac{(\alpha_I k D_P)^l}{l!}\right). \end{aligned} \tag{23}$$

**Proof.** Note that we can rewrite $P(C_{n,<k})$ as

$$\begin{aligned} P(C_{n,<k}) &= P\big(C_{n,<k}/B_{n,kD_P}\big) \times P\big(B_{n,kD_P}\big) + P\big(C_{n,<k}/\neg B_{n,kD_P}\big) \times P\big(\neg B_{n,kD_P}\big) = \\ &= P\big(C_{n,<k}/B_{n,kD_P}\big) \times P\big(B_{n,kD_P}\big) + 1 \times P\big(\neg B_{n,kD_P}\big), \end{aligned} \tag{24}$$

because $\neg B_{n,kD_P} \Rightarrow C_{n,<k}$.

Let us assume that $B_{n,kD_p}$ holds. Then, we can define the full group of events $\{H_l\}_{l=1}^n$ as

$H_l = \{$ exactly $l-1$ blocks out of $n-1$ blocks in SI blockchain were generated during the interval $(S_I(n) - kD_P; S_I(n))\}$, $l = \overline{1,n}$ (see Figure 1).

Note that $n$-th block was generated exactly at moment $S_I(n)$, because of the definition of $S_I(n)$. Then, under condition $B_{n,kD_P}$,

$$\begin{aligned} P(C_{n,<k}) &= \sum_{l=1}^{n-1} P(C_{n,<k}/H_l)P(H_l) + P(C_{n,<k}/H_n)P(H_n) < \\ &\sum_{l=1}^{n-1} P(C_{n,<k}/H_l)P(H_l) + P(H_n) < \\ &\sum_{l=1}^{n-1}\left(\sum_{i=0}^{k-1} C^i_{n-l+i-1} {p_P}^i {p_I}^{n-l}\right) \times e^{-k\alpha_l D_P} \times \frac{(k\alpha_I D_P)^{l-1}}{(l-1)!} + e^{-k\alpha_l D_P} \times \frac{(k\alpha_I D_P)^{n-1}}{(n-1)!}. \end{aligned}$$

If $B_{n,kD_P}$ doesn't hold, then $P(C_{n,<k}) = 1$.

Combining these two cases for $B_{n,kD_p}$ according to (26) and using (24), we obtain the statement (25) and the theorem is proved.

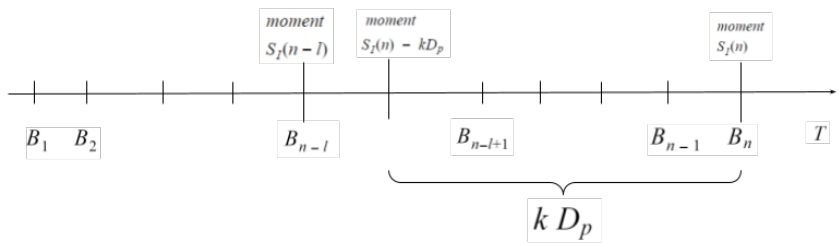


**Figure 1.** Event $H_1$.

Note that the expression (25) may also be used as an upper bound of $P(C_{n,<k})$ even in the case when $D_P = 0$. Thus, in this case, the first sum of (25) has only one nonzero term when $l = 1$. So we have

$$P(C_{n,<k}) < \sum_{i=0}^{k-1} C^i_{n-1+i-1} {p_P}^i {p_I}^{n-1}$$

which is the probability $P(C_{n-1,<k})$ which is large than $P(C_{n,<k})$.

Now we give some intuition for how we can use the result of Theorem 1.

Let the block of the deep $k$ in the SP blockchain be considered stable with some overwhelming probability $1 - \varepsilon$ for some sufficiently small $\varepsilon$. Let for the value $n \in \mathrm{N}$ the next statement hold: □

**Statement 1** (related to the number of blocks)**.** *"The probability, that during $n \in \mathrm{N}$ blocks were generated in the SI blockchain, not less than $k \in \mathrm{N}$ blocks were generated and shared in the SP blockchain, is not less than $1 - \delta$ for some sufficiently small $\delta$".*

According to the PoP concept, to guarantee stability in the SI blockchain for some block $B$ with an overwhelming probability $1 - \gamma$ for some given sufficiently small $\gamma$, we need to wait for such a number $k$ of blocks in the SP blockchain after the first reference to $B$, for which Statement 1 is true, with values $\varepsilon$ and $\delta$ such that $\varepsilon + \delta > \gamma$. Thus, the stability may fail in only two cases: when Statement 1 fails (with probability $\delta$ ) or when stability in the SP blockchain fails (with probability $\varepsilon$ ). Our main task is to find such $n \in \mathrm{N}$ that for a given sufficiently small $\delta$ and $k \in \mathrm{N}$, Statement 1 holds. This is the same as to find such $n \in \mathrm{N}$ that $P(C_{n,<k}) < \delta$.

## 3. Numerical Results

We calculate the probabilities of the event $C_{n,<k}$ according to (25) for different values $n$ from 10 to 400 with step 10 , for $k = 6$ (which is a common value for BTC), $\alpha_P = \frac{1}{600}, \alpha_I = \frac{1}{30}$, and for different values of $D_P = 0, 5, 10, 15, 20$ (Tables 1 and 2). Additionally, we give calculations for $n$ from 320 to 340 with step 1 in Table 3. Results from Tables 1–3 shows that the value $n_0(\delta, k)$ increases when $D_P$ increases. For example, as we can see from Table 3, with $\delta = 10^{-4}$ and $k = 6$, we obtain $n_0(\delta, k) = 335$ for $D_P = 0$, $n_0(\delta, k) = 337$ for $D_P = 5, n_0(\delta, k) = 338$ for $D_P = 10, n_0(\delta, k) = 339$ for $D_P = 15$ and $n_0(\delta, k) = 340$ for $D_P = 20$. Though the difference in these very cases is relatively small, we cannot ignore it in general. Note that it would be essentially large if the block generation rate $\alpha_P$ was large (here we take $\alpha_P = \frac{1}{600}$, as for BTC).

**Table 1.** Probabilities of $C_{n,<k}$ for $n = \overline{20, 200}$ with step 10 and $D_P = 0, 5, 10, 15, 20$.

| $nlD_p$ | 0 | 5 | 10 | 15 | 20 |
|---|---|---|---|---|---|
| 20 | 0.999058 | 0.999403 | 0.999527 | 0.999629 | 0.999712 |
| 30 | 0.994254 | 0.995711 | 0.996307 | 0.996837 | 0.997306 |
| 40 | 0.980719 | 0.984303 | 0.985876 | 0.987331 | 0.988673 |
| 50 | 0.953876 | 0.960391 | 0.963373 | 0.966203 | 0.968881 |
| 60 | 0.911192 | 0.920945 | 0.925539 | 0.929971 | 0.934241 |
| 70 | 0.852845 | 0.865602 | 0.871736 | 0.877724 | 0.883565 |
| 80 | 0.781395 | 0.796501 | 0.803877 | 0.811144 | 0.818299 |
| 90 | 0.700909 | 0.717481 | 0.725671 | 0.733798 | 0.741858 |
| 100 | 0.615995 | 0.633110 | 0.641650 | 0.650173 | 0.658674 |
| 110 | 0.531017 | 0.547845 | 0.556310 | 0.564797 | 0.573303 |
| 120 | 0.449588 | 0.465477 | 0.473522 | 0.481621 | 0.489770 |
| 130 | 0.374345 | 0.388844 | 0.396227 | 0.403683 | 0.411211 |
| 140 | 0.306932 | 0.319783 | 0.326358 | 0.333017 | 0.339760 |
| 150 | 0.248116 | 0.259223 | 0.264930 | 0.270724 | 0.276606 |
| 160 | 0.197971 | 0.207363 | 0.212206 | 0.217134 | 0.222148 |
| 170 | 0.156074 | 0.163864 | 0.167894 | 0.172003 | 0.176191 |
| 180 | 0.121689 | 0.128041 | 0.131337 | 0.134702 | 0.138139 |
| 190 | 0.093915 | 0.099016 | 0.101669 | 0.104383 | 0.107158 |
| 200 | 0.071798 | 0.075839 | 0.077946 | 0.080103 | 0.082313 |

**Table 2.** Probabilities of $C_{n,<k}$ for $n = \overline{210, 400}$ with step 10 and $D_P = 0, 5, 10, 15, 20$.

| $nlD_p$ | 0 | 5 | 10 | 15 | 20 |
|---|---|---|---|---|---|
| 210 | 0.054411 | 0.057573 | 0.059225 | 0.060919 | 0.062655 |
| 220 | 0.040902 | 0.043348 | 0.044628 | 0.045942 | 0.047292 |
| 230 | 0.030515 | 0.032388 | 0.033370 | 0.034380 | 0.035417 |
| 240 | 0.022606 | 0.024027 | 0.024773 | 0.025541 | 0.026331 |
| 250 | 0.016637 | 0.017706 | 0.018268 | 0.018847 | 0.019443 |
| 260 | 0.012170 | 0.012967 | 0.013387 | 0.013820 | 0.014266 |
| 270 | 0.008850 | 0.009441 | 0.009753 | 0.010074 | 0.010406 |
| 280 | 0.006402 | 0.006837 | 0.007066 | 0.007303 | 0.007547 |
| 300 | 0.003300 | 0.003531 | 0.003653 | 0.003780 | 0.003910 |
| 310 | 0.002353 | 0.002520 | 0.002608 | 0.002700 | 0.002794 |
| 320 | 0.001671 | 0.001791 | 0.001854 | 0.001920 | 0.001988 |
| 330 | 0.001181 | 0.001267 | 0.001313 | 0.001360 | 0.001409 |
| 340 | 0.000832 | 0.000893 | 0.000926 | 0.000960 | 0.000994 |
| 350 | 0.000584 | 0.000628 | 0.000651 | 0.000674 | 0.000699 |
| 360 | 0.000409 | 0.000439 | 0.000456 | 0.000472 | 0.000490 |
| 370 | 0.000285 | 0.000306 | 0.000318 | 0.000330 | 0.000342 |
| 380 | 0.000198 | 0.000213 | 0.000221 | 0.000230 | 0.000238 |
| 390 | 0.000137 | 0.000148 | 0.000153 | 0.000159 | 0.000165 |

**Table 3.** Probabilities of $C_{n,<k}$ for $n = \overline{330, 340}$ with step 1 and $D_P = 0, 5, 10, 15, 20$.

| $nlD_p$ | 0 | 5 | 10 | 15 | 20 |
|---|---|---|---|---|---|
| 330 | 0.001181 | 0.001267 | 0.001313 | 0.001360 | 0.001409 |
| 331 | 0.001141 | 0.001224 | 0.001268 | 0.001314 | 0.001361 |
| 332 | 0.001102 | 0.001182 | 0.001225 | 0.001269 | 0.001314 |
| 333 | 0.001064 | 0.001142 | 0.001183 | 0.001225 | 0.001270 |
| 334 | 0.001027 | 0.001102 | 0.001142 | 0.001183 | 0.001226 |
| 335 | 0.000992 | 0.001065 | 0.001103 | 0.001143 | 0.001184 |
| 336 | 0.000958 | 0.001028 | 0.001065 | 0.001104 | 0.001144 |
| 337 | 0.000925 | 0.000993 | 0.001029 | 0.001066 | 0.001104 |
| 338 | 0.000893 | 0.000958 | 0.000993 | 0.001029 | 0.001066 |
| 339 | 0.000862 | 0.000925 | 0.000959 | 0.000994 | 0.001030 |
| 340 | 0.000832 | 0.000893 | 0.000926 | 0.000960 | 0.000994 |

## 4. Discussion

We analyzed the block stability in the SI blockchain with the PoP consensus protocol and obtained results in a realistic model, with nonzero synchronization time, which are strictly mathematically proved and allow us to build security estimations of the SI blockchain. Using these results, we also can set the desirable security level for the SI blockchain and calculate the necessary number of confirmation blocks in it, which guarantee this security level, based on the SP blockchain like BTC.

The numerical results obtained using the formulas that are in this paper are completely in line with the expectations and show that the probability of an attack in the SI blockchain increases with increasing the synchronization time in the SP blockchain and, accordingly, more confirmation blocks are required for protection against this attack. It should also be taken into consideration that when the block generation intensity in the SP blockchain increases, the probability of an attack will increase faster with increasing synchronization time.

These results may also be useful for building secure Altchains, which use VeriBlock blockchain as the SP blockchain, if both blockchains are based on a Proof-of-Work protocol.

Note that if at least one of these two blockchains—the SI blockchain or SP blockchain—is based on some other protocol, such as Proof-of-Stake, one must use a completely different approach to build security estimations of such blockchains.

The idea behind the PoP protocol, as well as similar ideas from earlier articles mentioned in the review, allows us not only to ensure the resilience of new blockchains that initially have a small PoW value in the block, but also to build security blockchains with high throughput using any of the existing blockchains with provable security. However, it does not solve the second significant problem of blockchains—reducing block confirmation time, i.e., the time until the moment when transactions in the block may be considered uninvertible with overwhelming probability. Maybe the described problem can be solved, at least partially, if we manage to generalize the PoP consensus to the case of SI DAGchain instead of the blockchain. This approach may be the next direction of further investigations.

**Author Contributions:** Conceptualization, L.K. and V.K.; methodology, L.K.; numerical results, O.P. and O.M.; validation, O.P. and O.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| PoW | Proof-of-Work Consensus Protocol |
| PoS | Proof-of-Stake Consensus Protocol |
| PoP | Proof-of-Proof Consensus Protocol |
| MC | MainChain |
| SC | SideChain |
| VB | Veriblock |
| SP | Security Provided |
| SI | Security Inherited |
| RV | Random Variable |
| tps | Transactions per Second |

## References

1. Proof-of-Proof and VeriBlock Protocol Consensus and Economic Incentivization Specifications'. Available online: https://veriblock.org/wp-content/uploads/2019/06/Proof-of-Proof_and_VeriBlock_Blockchain_Protocol_Consensus_Algorithm_and_Economic_Incentivization_v1.0.pdf (accessed on 1 June 2019).
2. Sompolinsky, Y.; Zohar, A. Secure High-Rate Transaction Processing in Bitcoin. In *Financial Cryptography and Data Security. FC 2015*; Böhme, R., Okamoto, T., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2015; Volume 8975. [CrossRef]
3. Sompolinsky, Y.; Zohar, A. Accelerating bitcoin's transaction processing: Fast money grows on trees, not chains. *IACR Cryptol. Eprint Arch.* **2013**, *881*, 2013.
4. Sompolinsky, Y.; Lewenberg, Y.; Zohar, A. SPECTRE: A Fast and Scalable Cryptocurrency Protocol. *IACR Cryptol. Eprint Arch.* 2016, 1159. Available online: https://eprint.iacr.org/2016/1159 (accessed on 30 April 2018).
5. Sompolinsky, Y.; Wyborski, S.; Zohar, A. PHANTOM and GHOSTDAG: A Scalable Generalization of Nakamoto Consensus. Cryptology ePrint Archive, Paper 2018/104. 2018. Available online: https://eprint.iacr.org/2018/104 (accessed on 30 April 2018).
6. Popov, S. The Tangle. Available online: https://iota.org/IOTA (accessed on 30 April 2018).
7. Whitepaper Prism. Available online: https://pzm.space/prizm-whitepaper/ (accessed on 1 June 2020).
8. Fitzi, M.; Gaži, P.; Kiayias, A.; Russell, A. Parallel Chains: Im-Proving Throughput and Latency of Blockchain Protocols via Parallel Composition. *Cryptol. Eprint Arch.* Available online: https://eprint.iacr.org/2018/1119 (accessed on 30 April 2018).
9. Pass, R.; Shi, E. FruitChains: A fair blockchain. In *36th ACMPODC*; Schiller, E.M., Schwarzmann, A.A., Eds.; ACM: New York, NY, USA, 2017; pp. 315–324
10. Garoffolo, A.; Kaidalov, D.; Oliynykov, R. Zendoo: A zk-SNARK verifiable cross-chain transfer protocol enabling decoupled and decentralized sidechains. In Proceedings of the 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS), Singapore, 29 November–1 December 2020.
11. Bagaria, V.; Kannan, S.; Viswanath, P. Prism: Deconstructing the Blockchain to Approach Physical Limits. In Proceedings of the CCS '19: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, London, UK, 6 November 2019; pp. 585–602.
12. Pinzon, C. Double-Spend Attack Models with Time Advantage for Bitcoin. *Electron. Notes Theor. Comput. Sci.* **2016**, *329*, 79–103. [CrossRef]
13. Gazi, P.; Kiayias, A.; Russell, A. Tight consistency bounds for bitcoin. In Proceedings of the 2020 ACM SIGSAC Conference on Compute and Communications Security, Virtual, 9–13 November 2020; pp. 819–838.
14. Kovalchuk, L.; Kaidalov, D.; Nastenko, A.; Rodinko, M.; Shevtsov, O.; Oliynykov, R. Decreasing security threshold against double spend attack in networks with slow synchronization. *Comput. Commun.* **2020**, *154*, 75–81. [CrossRef]
15. Kovalchuk, L.; Kaidalov, D.; Nastenko, A.; Rodinko, M.; Oliynykov, R. Probability of double spend attack for network with non-zero synchronization time. In Proceedings of the 21th Central European Conference on Cryptology CECC 2021, Debrecen, Hungary, 23–25 June 2021; pp. 52–54.
16. Grunspan, C.; Pérez-Marco, R. Double Spend Races. CoRR abs/1702.02867. Available online: http://arxiv.org/abs/1702.02867 (accessed on 1 June 2017).