The Evaluation of a Double-Spend Attack Probability for Ouroboros-Like Proof-of-Stake Consensus

Lyudmila Kovalchuk , Mariia Rodinko , Roman Oliynykov , Volodymyr Artemchuk *

Posted Date: 10 April 2026

doi: 10.20944/preprints202604.0694.v1

Keywords: blockchain; Proof-of-Stake; consensus protocol; double-spend attack; Ouroboros protocol

 doi:10.20944/preprints202604.0694.v1

Article

The Evaluation of a Double-Spend Attack Probability for Ouroboros-Like Proof-of-Stake Consensus

Lyudmila Kovalchuk [1,2,3], Mariia Rodinko [4], Roman Oliynykov [4] and Volodymyr Artemchuk [1,5,6,*]

[1] G. E. Pukhov Institute for Modelling in Energy Engineering of the National Academy of Sciences of Ukraine, Kyiv, Ukraine
[2] State Scientific and Research Institute of Cybersecurity Technologies and Information Protection, Kyiv, Ukraine
[3] National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute», Kyiv, Ukraine
[4] V. N. Karazin Kharkiv National University, Kharkiv, Ukraine
[5] Center for Information-Analytical and Technical Support of Nuclear Power Facilities Monitoring of the National Academy of Sciences of Ukraine, Kyiv, Ukraine
[6] Kyiv National Economic University named after Vadym Hetman, Kyiv, Ukraine
* Correspondence: volodymyr.artemchuk@pimee.ua

Abstract

This paper studies the probability of a double-spend attack in an Ouroboros-like Proof-of-Stake (PoS) setting when confirmation decisions must be made for a finite number of blocks. Existing security analyses of Ouroboros-family protocols are mainly asymptotic and therefore do not directly provide the attack probability for a fixed confirmation depth. We consider an analytically tractable model that allows empty slots and multiple slot leaders, and assumes fixed stake distribution within an epoch, one-block growth of the public longest chain in any slot containing at least one honest leader, and next-slot block visibility. These assumptions hold when the time slot length is much greater than the network delay, and are applicable to practical deployment scenarios such as Cardano. Under these assumptions, for the first time, an exact closed-form solution for the success probability of a double-spend attack considering a realistic model with multiple leaders and empty time slots. Numerical examples illustrate how the required confirmation depth depends on the adversarial stake ratio and the active slot coefficient. The results apply to the stated analytical model and do not yet cover delayed fork resolution or the full protocol-level fork-choice and finality mechanisms of Ouroboros Praos.

Keywords: blockchain; Proof-of-Stake; consensus protocol; double-spend attack; Ouroboros protocol

1. Introduction

Blockchain technology continues to gain ground in various fields such as cryptocurrencies, logistics, healthcare, education, copyright, and others. Its implementation in these areas ensures security, reliability, and data transparency. Interest in blockchain increased during the COVID-19 pandemic as well. Despite the many advantages of blockchain technology, some problems remain, such as transaction confirmation time and high implementation costs [1]. Therefore, research in this area remains highly relevant.

Blockchain technology has immense potential to revolutionize the FinTech industry [2]. Since the introduction of Bitcoin in 2009, the industry has evolved significantly, including the creation of Ethereum, Cardano, Polkadot, and other prospective cryptocurrencies. Ensuring the security of transactions is a crucial responsibility for these systems.

Recent advancements in decentralized Internet of Things (IoT) architectures increasingly rely on blockchain technology to ensure data integrity, device autonomy, and secure micro-transactions. Proof-of-Stake (PoS) consensus protocols, particularly Ouroboros-like variants, are highly favored in resource-constrained IoT environments due to their energy efficiency and scalability. However, the security of these underlying consensus mechanisms against attacks like double-spending, especially in

 doi:10.20944/preprints202604.0694.v1

the presence of realistic network conditions and time delays inherent to large-scale IoT deployments, remains a critical concern.

Recent studies also confirm that blockchain-assisted IoT and fog environments require robust trust, authentication, and cybersecurity mechanisms, which further motivates rigorous analysis of consensus-level attack resistance [3**?** ,4].

The probability of success of a double-spend attack for Proof-of-Work (PoW), depending on the number of confirmation blocks, was calculated by Nakamoto for Bitcoin [5]. Since then, estimates for this attack have been refined by various authors. The corresponding estimates were also obtained for Proof-of-Stake consensus.

For cryptocurrency, a double-spend attack means that the attacker tries to spend the same coin twice, creating an alternative chain with an alternative transaction [5]. An exploit typically targets blockchain networks, especially those with weak consensus mechanisms or low network security. The steps of a double-spend attack are the following.

1. **Transaction Initiation (honest spend)**.
   - The attacker creates a valid transaction and sends it to a merchant or recipient.
   - The transaction is broadcast to the network and appears in the mempool (unconfirmed transactions).
   - The recipient sees the transaction as pending and may accept it if they trust the network.
2. **Broadcasting an alternative transaction (malicious spend)**.
   - The attacker creates a second transaction that spends the same coins but directs them to their own wallet instead of the merchant.
   - This transaction has the same inputs as the first one but different outputs.
   - The attacker does not broadcast it immediately, waiting for the right moment.
3. **Gaining an advantage in mining (or forging)**. Depending on the blockchain's consensus mechanism, the attacker uses different strategies. In PoW chains (e.g., Bitcoin):
   - The attacker controls a significant portion of the network's hash power (usually 50% or more).
   - They mine a private fork of the blockchain, keeping their double-spend transaction hidden.

   In PoS chains:
   - The attacker uses a high stake ratio or exploits weaknesses in the forking rules.
   - They may attempt a long-range attack or manipulate finality assumptions.
4. **Overwriting the honest transaction**.
   - If the attacker's private chain grows longer than the public chain, it becomes the valid one under the longest chain rule.
   - The network reorganizes, and the attacker's conflicting transaction is included, while the original transaction to the merchant is dropped.
   - The merchant, thinking the transaction was valid, may have already delivered goods or services.
5. **Profit and network impact**.
   - The attacker retrieves the funds and retains both the goods/services and their original balance.
   - The attack damages network trust, causing users to lose confidence in the blockchain.

The idea is also presented in Figure 1 $tx1$ is the initial honest transaction, and $tx2$ is the alternative (conflicting) malicious transaction.

To mitigate the risk of an attack, the merchant waits for a specific number of confirmation blocks (e.g., 6 blocks in Bitcoin). Our results precisely determine how many blocks must be confirmed to reduce the probability of an attack to the desired level.

 doi:10.20944/preprints202604.0694.v1

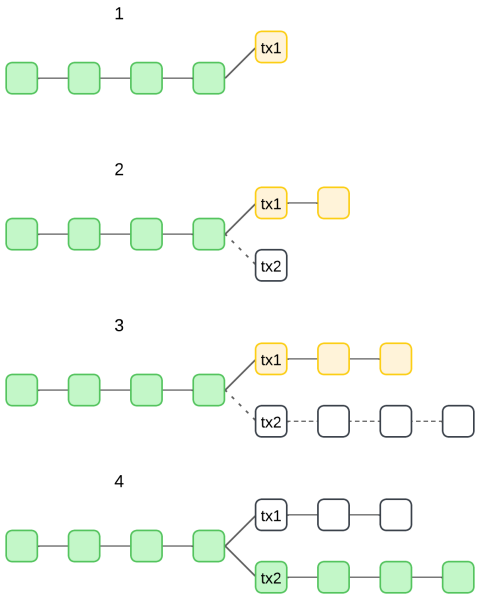

Figure 1. The steps of a double-spend attack.

Note that forks may also occur because of the time required for network synchronization. In this case, the vendor, before sending goods, has to make sure that his transaction is included in all visible longest chains, and all such chains, the number of confirmation blocks is not less than the recommended one. Under "number of confirmation blocks after transaction," we mean the minimum number of confirmation blocks among all visible longest chains if there are multiple longest chains.

A double-spend attack is also interesting regarding blockchains with PoS consensus protocols. The main idea of PoS consensus is randomized slot leader selection; that is, a participant who forges the next block is randomly selected in a non-biased way to issue the block within a given time (timeslot) [6]. Slot leaders (SLs) are chosen among stakeholders with probabilities proportional to their stakes.

Such an approach is implemented in the Ouroboros protocol – the first provably secure PoS consensus protocol. Its original version called Ouroboros Classic was presented in [7]. Then several improved versions appeared: Ouroboros Praos [8] (security against fully adaptive corruption in the semi-synchronous model), Ouroboros Genesis [9] (security with a dynamic participation model), Ouroboros Chronos [10] (a provable secure PoS consensus protocol that is independent of global time). However, all security estimations in these articles are asymptotic and can't be applied to obtain numerical results, like the probability of attacks, for a fixed number of confirmation blocks.

For example, the original Ouroboros paper [7] proves security only via asymptotic persistence, liveness, common-prefix, chain-growth, and chain-quality bounds and does not provide an explicit formula for the success probability of a double-spend attack as a function of the confirmation depth. In contrast, our paper derives explicit analytic expressions that allow this probability to be computed for a fixed confirmation depth and therefore support practical selection of the required number of confirmation blocks.

Ouroboros processes blocks by dividing the blockchain into epochs, which are divided into timeslots. In Ouroboros, block production is not a race between stakeholders. Instead, a slot leader (SL) is randomly chosen on the basis of the number of tokens they own (their stake). In reality, there may be multiple SLs per timeslot, or some slots may be empty.

In the PoS protocol, only stakeholders with some essential stake may become SLs and create blocks in corresponding time periods, named timeslots. In what follows, we consider two types of stakeholders/SLs: honest and malicious. We assume that honest SLs act according to the general blockchain rules: collect all new transactions in a block; add this new block to the longest chain; and immediately share the new block between all nodes. Malicious SLs can break any rule. They can add a

 doi:10.20944/preprints202604.0694.v1

newly created block to any of the existing chains, delay its sharing for some nodes on arbitrary time, or even create one of the hidden chains (alternative chain) of sufficient length without sharing it with honest nodes, and then, if it becomes the longest, share this whole chain with all nodes at once. Also, they may make censorship attacks without including some transactions in the blocks they create.

In [6], the analytical estimations of a double-spend attack in the covert adversary model are presented for an arbitrary version of the Ouroboros protocol, but for a very simplified model. In that model, it is assumed that there are "honest" and "malicious" timeslots, and particular participants use "their" timeslots to produce a block.

As we can see from the review of papers (e.g., [8,11–13]), many scientists are actively researching consensus protocols for blockchains and their security against various attacks. Despite this, as of now, questions remain open regarding obtaining accurate explicit (not asymptotic) estimates of the probability of a double-spending attack success, obtained for the realistic model of the Ouroboros protocol. The relevance and importance of such estimates are enhanced by the fact that this protocol is used in many blockchains (Cardano, Mina, Polkadot, Horizen sidechains), and their users need to understand when a block with their transaction can be considered finalized.

In this paper, we analyze a more realistic than prior, but still simplified, Ouroboros-like PoS model. In particular, we allow empty slots and multiple slot leaders, while assuming next-slot synchronization, fixed stake distribution within an epoch, and one-block growth of the public longest chain in every slot with at least one honest leader. These assumptions make it possible to obtain explicit non-asymptotic estimates for the success probability of a double-spend attack and for the confirmation depth required to reduce this probability below a prescribed threshold. The SL distribution in Ouroboros depends on *active slot coefficient* f, introduced in [8] and described below. According to the longest chain rule, honest SL always add the newly created block to the longest chain (or to one of the longest, if there are several longest chains of the same length) and share this block instantly. Because of this, we assume that after each timeslot with at least 1 honest SL, the longest chain grows with 1 block. We also make the simplified assumption about instant synchronization, i.e., the block issued in some timeslot is seen for all nodes at the beginning of the next timeslot. It is natural to make this assumption when the duration of the timeslot is significantly longer than the time delay, or when the value of f is small. In these cases, the average distance between nonempty timeslots is significantly larger than the time delay, and these scenarios are the focus of this work. Based on these assumptions, we obtained formally proven analytical results for the probability of a double-spend attack for the Ouroboros-based PoS protocol. The obtained formulas not only allow us to calculate the probability of a successful attack based on network parameters and the number of confirmation blocks, but also help us to find the sufficient number of confirmation blocks for which the probability of an attack is negligible. The formulas are validated by the corresponding numerical results and graphs.

The main difference of our paper from previous papers that analyze the Ouroboros protocol [8,11–13] is that in these papers only asymptotic estimates were obtained, which cannot be used to calculate a certain number of blocks that guarantee that the probability of attack is less than the given value. Here we give accurate estimates and explicit formulas which may be used for both calculating the probability of a successful attack and the sufficient number of confirmation blocks. Note that the paper [6] also gives explicit formulas for the probability without asymptotics, but, as was already mentioned, for a simplified model, when each timeslot has exactly one slot leader. We propose a model without such simplifications that allows one to obtain accurate estimates of the probability of a double-spend attack on the Ouroboros-like consensus protocol for different network parameters and different stake ratios of the adversary.

The **practical benefit** of the results obtained is the following: it is possible to calculate how many confirmation blocks should be created after a block with a certain transaction so that this block cannot be canceled with a given probability. This can be useful both for protocol developers, when choosing network parameters, for example, the intensity of block generation, and for vendors who want to be sure that they will not become victims of attackers.

 doi:10.20944/preprints202604.0694.v1

In IoT scenarios, device resources are limited and network latency fluctuates greatly. The instant synchronization hypothesis in this article needs further verification. Future work may consider analyzing the attack probability under partially synchronized models.

The paper is organized as follows. In Section 2, we provide a literature review on evaluating the probability of a double-spend attack on consensus protocols. In Section 3, we present several statements necessary for our investigation and key designations introduced in the Ouroboros protocol, briefly describe the formal attack setup and the model scope, and then formulate the main result of this paper – an explicit formula for calculating the probability of a double-spend attack on Ouroboros-like consensus protocols for real applications. In Section 4, we present numerical results obtained using the derived formulas. In Section 5, we discuss the results. Finally, in Section 6, we draw conclusions.

2. Related Work

The literature relevant to this paper can be divided into three groups: studies of double-spend probabilities in PoW systems, broader works on PoS and blockchain security, and Ouroboros-specific analyses.

The concept of a double-spend attack and its prevention using PoW, along with the notion of confirmation blocks, was first introduced by S. Nakamoto in [5]. To ensure protection against this attack on Bitcoin, Nakamoto proposed not to supply goods as soon as the transaction occurred, but to wait for some time, more precisely for several confirmation blocks, and only then, if the transaction has not disappeared from the blockchain, to supply goods [6]. Nakamoto also proposed a method for PoW consensus to calculate the number of blocks a supplier should wait to ensure the transaction is irreversible. However, this method has significant shortcomings, such as applying Poisson approximation for negative binomial distribution in the cases when such approximation gives incorrect results, etc.

The probability of a double-spend attack was further analyzed in [14,15], but the models presented in these papers also have some unproven statements. The paper [16] was the first to formally prove that the probability of a double-spend attack for PoW consensus has a negative binomial distribution. For the first time, it has been proven that as the number of confirmation blocks grows, the probability of a fork decreases exponentially. The authors, however, also based their model on the assumption of no delay in block propagation. The paper [17] generalizes results obtained in [16]. In this work, the authors provide a formally proven expression for the probability of double-spending attacks in a model for PoW with continuous time and without assumptions about discrete timeslots. The expression considers network parameters, including network synchronization time. The paper [18] investigates the profitability of double-spend attacks, providing an example of a profitable attack against BitcoinCash.

The second group of works concerns PoS security more broadly. There is extensive literature on PoS protocol design, comparative analysis, sharding, and adversarial models. For example, the paper [19] introduces sharding-based PoS Blockchain protocols combining PoS and practical Byzantine Fault Tolerance consensus mechanisms. The paper [20] proposes a new Robust PoS consensus protocol, which limits the maximum value of the coin age to effectively avoid coin age accumulation attacks and Nothing-at-Stake attacks. The paper [21] proposes a process that allows a node to join or leave as a validator in a PoS-based blockchain network, enhancing the overall security of the main chain consensus process. The paper [22] introduces SMPTC3, an innovative approach specifically designed to enhance security and privacy in cross-chain transaction verification. Furthermore, a cross-shard transaction processing protocol called cross-reserve is proposed in [23]. It removes the need for costly cross-shard coordination while ensuring the same consistency and atomicity guarantees. The paper [24] examines different aspects of Cardano economics, providing a comprehensive analysis of the Cardano blockchain, with a focus on the Ouroboros protocol and its impact on entity wealth and stake delegation dynamics within the ecosystem. Many papers also focus on the analysis of PoW and PoS, comparing such parameters as power consumption, security, scalability, and decentralization [25–31].

 doi:10.20944/preprints202604.0694.v1

These works are important for understanding the wider design space, but they do not directly provide explicit confirmation-depth formulas for a finite-block settlement problem.

The third and most relevant group is the Ouroboros line of research. Ouroboros Classic [7],Ouroboros Praos [8], Ouroboros Genesis [9], Ouroboros Chronos [10], and more recent related analyses [12,13] establish provable security guarantees for Ouroboros-style protocols under different assumptions. These results are foundational, but they are mainly asymptotic and are not designed to return a direct numerical attack probability for a fixed confirmation depth.

The closest work to the present paper is [6], which provides explicit non-asymptotic formulas for a covert-adversary model related to Ouroboros. However, that analysis assumes a substantially simplified setting in which each slot has exactly one slot leader. In contrast, real Ouroboros-like settings may include both empty slots and multiple slot leaders.

Therefore, the contribution of this paper is not to replace the full protocol-level security theory of Ouroboros, but to fill a narrower and practically important gap: we derive an explicit non-asymptotic confirmation-depth formula for an analytically tractable Ouroboros-like model that extends the one-slot-leader abstraction by allowing empty slots and multiple slot leaders under clearly stated synchronization and chain-growth assumptions.

3. Materials and Methods

In this section we formulate and prove the main properties of the PoS protocol and other useful statements, based on which we then prove the main results regarding the probability of success of a double-spend attack.

3.1. Ouroboros-like Slot Model and Properties of the Block Creation Function

In this subsection, we briefly describe some designations and main ideas introduced in the Ouroboros protocol [7].

We define S_i, $i \in I = \{1, \dots, n\} = I_H \cup I_M$, the set of all stakeholders with corresponding stake ratios α_i, $\sum_{i=1}^{n} \alpha_i = 1$ (here I_H / I_M are subsets of indexes corresponding to honest/malicious stakeholders).

Following the definitions and designations, introduced in [8], we define *active slot coefficient (ASC)* f and corresponding function $\varphi_f(\alpha) = 1 - (1-f)^{\alpha}$, which we will call *block creation function (BCF)*, where $f \in (0, 1)$.

Proposition 1 (properties of BCF). *In our designations BCF has the next properties for* $\alpha_i \geq 0, i = 1, 2$ *:*

1. $\varphi_f(\alpha_1) + \varphi_f(\alpha_2) - \varphi_f(\alpha_1) \cdot \varphi_f(\alpha_2) = \varphi_f(\alpha_1 + \alpha_2)$;
2. $\varphi_f(\alpha_1) + \varphi_f(\alpha_2) \geq \varphi_f(\alpha_1 + \alpha_2)$.

Proof. Property 1 can be obtained through direct algebraic verification and definition of $\varphi_f(\alpha) = 1 - (1-f)^{\alpha}$:

$$\begin{aligned}
&\varphi_f(\alpha_1) + \varphi_f(\alpha_2) - \varphi_f(\alpha_1) \cdot \varphi_f(\alpha_2) = \\
&= 1 - (1-f)^{\alpha_1} + 1 - (1-f)^{\alpha_2} - (1 - (1-f)^{\alpha_1})(1 - (1-f)^{\alpha_2}) = \\
&= 1 - (1-f)^{\alpha_1} + 1 - (1-f)^{\alpha_2} - 1 + (1-f)^{\alpha_1} + (1-f)^{\alpha_2} - (1-f)^{\alpha_1}(1-f)^{\alpha_2} = \\
&= 1 - (1-f)^{\alpha_1}(1-f)^{\alpha_2} = 1 - (1-f)^{\alpha_1 + \alpha_2} = \varphi_f(\alpha_1 + \alpha_2).
\end{aligned}$$

Property 2 is derived from Property 1 and the fact that the value $\varphi_f(\alpha_1) \cdot \varphi_f(\alpha_2)$ is positive.

The peculiarity of the function is that, on the one hand, this function φ is little different from linear, as is necessary for a fair distribution of slot leaders, and, on the other hand, it protects against both the creation of coalitions and Sybil attacks, since these actions do not lead to any advantages in the distribution of timeslots.

Note 1: according to the Ouroboros protocol, for each stakeholder S_i with stake ratio α_i, the probability of being an SL in any timeslot is equal to

 doi:10.20944/preprints202604.0694.v1

$$p_i = 1 - (1 - f)^{\alpha_i}.$$

Note 2: according to Ouroboros slotleader election protocol, events "Stakeholder S_i is slotleader in timeslot T_k " and "Stakeholder S_j is slotleader in timeslot T_l " are independent.

Note 3: due to the first property of BCF, for any two stakeholders S_i and S_j with stake ratios α_i and α_j, the probability that at least one of them is SL in some timeslot with number l is equal to the probability that some stakeholder with stake $\alpha_i + \alpha_j$ is SL in this slot. In particular, it means that it doesn't matter if several stakeholders decide to unite their stakes or act separately; they have no advantages in any of these cases.

Note 4: in these assumptions, the probability that the timeslot isn't empty is equal to

$$p = 1 - (1 - f)^1 = f,$$

and the probability of the inverse event (TS is empty) is equal to $1 - f$.

Define α_H total stake ratio of honest stakeholders and $\alpha_M = 1 - \alpha_H$ total stake of malicious ones.

For each timeslot (TS) $T_i, i \geq 1$, introduce the next events:

- event $H\bar{M}$ = "all SLs in TS T_i are honest";
- event $\bar{H}M$ = "all SLs in TS T_i are malicious";
- event HM = "among SLs in TS T_i, both honest and malicious SLs are present";
- event C = "TS T_i is empty (there are no SLs in this TS)";
- event $D = HM \cup \bar{H}M \cup H\bar{M}$ = "TS T_i is not empty (there is at least one SL in this TS)".

From the definition of BCF and its properties, and considering Note 1 and Note 2, we immediately get the next Proposition, which is rather obvious and is given here without proof.

All formulas derived in Propositions 2–6 have rigorous analytical proofs based on a formal mathematical model and are obtained without using any empirical methods. In particular, we use the Ouroboros block creation function of the form $\varphi_f(\alpha) = 1 - (1 - f)^\alpha$, according to which slot leaders are selected and which gives the probability that a stakeholder with stake share α becomes a slot leader in a given slot. Accordingly, for the total normalized stake $\alpha = 1$, the probability that a slot is non-empty is f, while the probability of the opposite event, namely that the slot is empty, is $1 - f$. In proving the statements of Propositions 2 and 3, we also use the fact that the events related to slot-leader selection are independent, both for different slots and for different stakeholders within the same timeslot.

Proposition 2. *In our designations and assumptions, the next equalities hold:*

$$\begin{aligned} P(H\bar{M}) &= \varphi(\alpha_H)(1 - \varphi(\alpha_M)) = (1 - (1 - f)^{\alpha_H}) \cdot (1 - f)^{\alpha_M}; \\ P(\bar{H}M) &= \varphi(\alpha_M)(1 - \varphi(\alpha_H)) = (1 - (1 - f)^{\alpha_M}) \cdot (1 - f)^{\alpha_H}; \\ P(HM) &= \varphi(\alpha_H) \cdot \varphi(\alpha_M) = (1 - (1 - f)^{\alpha_H}) \cdot (1 - (1 - f)^{\alpha_M}); \\ P(C) &= 1 - f; \\ P(D) &= f. \end{aligned} \tag{1}$$

3.2. Formal Attack Setup

The general mechanics of a double-spend attack were introduced in Section 1. In the analytical model used below, we consider a vendor that accepts a transaction and waits until the block containing this transaction receives z confirmation blocks. During that period, the attacker privately maintains an alternative chain containing a conflicting transaction.

If, before the vendor releases goods or services, the attacker's private chain already becomes longer than the public chain with the original transaction, the attack succeeds immediately. Otherwise, after the vendor acts on the z-th confirmation block, the attacker may continue extending the hidden chain and attempt to catch up later. Our analysis therefore consists of two phases: the pre-delivery

 doi:10.20944/preprints202604.0694.v1

phase, during which the attacker accumulates a private-chain deficit, and the catch-up phase, during which the attacker attempts to eliminate this deficit.

3.3. Model Scope and State-Space Reduction

We mostly use a method of modeling for the description of a model under investigation, and methods of probability theory. Each process, connected with blockchain (slot leader elections, block creation, catching up the longest chain, etc.), we describe as corresponding random event, a variable, or process, or a composition of random variables. Then we formulate the event "DS attack was successful" in terms of these random variables and events. This allows one to represent our model using language and methods of probability theory and random processes, and obtain explicit formulas for the probability of a successful attack.

We assume that the stake distribution does not change during the epoch. This is standard in many analyses of PoS blockchain security and is especially natural when slot leaders are determined for an epoch in advance [8]. We also assume next-slot visibility of newly created blocks and the following longest-chain abstraction: in every slot containing at least one honest slot leader, the public honest longest chain grows by one block.

These assumptions define the scope of the analytical model studied in this paper. The model is *Ouroboros-like*, but it is not intended to reproduce the full protocol-level dynamics of Ouroboros Praos. In particular, we do not model the full fork tree, the density rule, rollback bounds, finality mechanisms, cross-epoch dependence, or adversarial influence on epoch randomness. Instead, we study only the evolution of two chain lengths: the public honest longest chain and the adversary's best hidden chain.

Under the adopted assumptions, an event of type $H\bar{M}$ increases the public-chain length by one block; an event of type $\bar{H}M$ increases the adversary's private-chain length by one block; an event of type HM increases both by one block; and an empty slot changes neither chain length. Consequently, for the attack analysis, the relevant state can be reduced to a one-dimensional block deficit between the adversary's best hidden chain and the public honest longest chain.

As we proceed, we will use the following conditional probabilities for a *non-empty* slot:

$$p_H = P(H\bar{M} \mid D), \qquad p_M = P(\bar{H}M \mid D), \qquad p_{HM} = P(HM \mid D).$$

The main differences of the model, considered in this paper, are the following:

- existence of empty timeslots;
- possibility to get several SLs in one timeslot, where each of them may create a valid block and add it to one of the existing chains (an honest SL always adds a newly created block to one of the longest chains).

As we proceed, we will use the following probabilities:
p_H – the probability that all SLs in a non-empty timeslot are honest;
p_M – the probability that all SLs in a non-empty timeslot are malicious;
p_{HM} – the probability that in a non-empty timeslot there are both honest and malicious SLs.

Proposition 3. *For the values p_H, p_M and p_{HM}, the next equalities hold:*

$$p_H = \frac{\varphi(\alpha_H)(1-\varphi(\alpha_M))}{f}; p_M = \frac{\varphi(\alpha_M)(1-\varphi(\alpha_H))}{f}; p_{HM} = \frac{\varphi(\alpha_H)\varphi(\alpha_M)}{f}. \tag{2}$$

Proof. We may write the probabilities p_H, p_M, and p_{HM} as the corresponding conditional probabilities, and calculate them according to the definition:

$$p_H = P(H \cap \bar{M}/D) = \frac{P(H \cap \bar{M} \cap D)}{P(D)} = \frac{P(H \cap \bar{M})}{P(D)} = \frac{\varphi(\alpha_H)(1-\varphi(\alpha_M))}{f};$$
$$p_M = P(\bar{H} \cap M/D) = \frac{P(\bar{H} \cap M \cap D)}{P(D)} = \frac{P(\bar{H} \cap M)}{P(D)} = \frac{\varphi(\alpha_M)(1-\varphi(\alpha_H))}{f};$$

 doi:10.20944/preprints202604.0694.v1

$$p_{HM} = P(H \cap M / D) = \frac{P(H \cap M \cap D)}{P(D)} = \frac{P(H \cap M)}{P(D)} = \frac{\varphi(\alpha_H)\varphi(\alpha_M)}{f}.$$

Note that for small enough coefficient f (such that $f << 1$) and for $x \in [0,1]$ we have $\varphi(x) = 1 - (1-f)^x \approx 1 - (1 - x \cdot f) = x \cdot f$. In this case, we can approximate

$$p_H \approx \frac{\alpha_H \cdot f \cdot (1 - \alpha_M \cdot f)}{f} = \alpha_H \cdot (1 - \alpha_M \cdot f), p_M \approx \alpha_M \cdot (1 - \alpha_H \cdot f),$$
$$\text{and } p_{HM} \approx \frac{\alpha_H \cdot f \cdot \alpha_M \cdot f}{f} = \alpha_H \cdot \alpha_M \cdot f,$$

or even $p_H \approx \alpha_H, p_M \approx \alpha_M, p_{HM} \approx 0$, if f is very close to zero.

3.4. The Probability of a Double-Spend Attack for Ouroboros-like Consensus Protocols

In this Section, we formulate the main result of this paper – the explicit formula for the calculation of the probability of a double-spend (DS) attack for Ouroboros-like consensus protocols in real applications.

For each $z \in \mathrm{N}$ introduce Random Variable (RV) $S(z)$, equal to the number of blocks, which malicious SLs create on some interval with fixed left edge till the TS, in which honest SL create z-th confirmation block (after block with the transaction which is under attack). So the event

$$S(z) = k$$

means that malicious SLs create exactly k blocks till the TS, in which the honest SL creates z-th block (starting from the time when the block containing the target transaction is broadcasted, count the time slots between then and the generation of the z-th confirmation block).

Proposition 4. *The probability distribution $P_z(k), k \in \mathrm{N} \cup \{0\}$, of the RV $S(z)$, is described with the next equalities:*

$$P_z(k) = \sum_{t=0}^{\min\{z,k\}} C_{z+k-t-1}^{z-1} \cdot C_z^t \cdot {p_M}^{k-t} \cdot {p_H}^{z-t} \cdot {p_{HM}}^t.$$

Proof. Define $|HM|, |\bar{H}M|, |H\bar{M}|$ as the number of corresponding events that happened after the block with the transaction under attack and till the z-th confirmation block, created by the honest miners. Note that malicious miners don't add blocks to the chain with the initial transaction (with payment to a vendor); they add blocks only to the alternative chain with an alternative transaction. In the case of exactly z confirmation blocks (after the block with the initial transaction), we have $|HM| + |H\bar{M}| = z$. If malicious stakeholders create exactly k blocks till this time in their hidden (alternative) chain, then $|HM| + |\bar{H}M| = k$. In this case the value of $|HM|$ may take any integer value t for $0 \le t \le \min\{k, z\}$. When t is fixed, then $|H\bar{M}| = z - t$ and $|\bar{H}M| = k - t$.

Fix some appropriate t and find the probability of the event $U(z,k,t)$, which consists of all such possible sequences of events $H\bar{M}, \bar{H}M$, and HM, where

$$\{|HM| + |H\bar{M}| = z \cap |HM| + |\bar{H}M| = k \cap |HM| = t$$
$$\textit{and the last event in the sequence is from the set } HM \cup H\bar{M}\}.$$

With fixed z, k, and t, the sequence length is $z + k - t$. As one event from the set $HM \cup H\bar{M}$ must be the last in the sequence, then all other $z - 1$ events from this set can be placed on $z + k - t - 1$ places in the sequence in $C_{z+k-t-1}^{z-1}$ ways. Next, among z places occupied with events $HM \cup H\bar{M}$, we can choose C_z^t places for HM. And all other $k - t$ places are occupied with events $\bar{H}M$.

So the probability of $U(z,k,t)$ is

$$P(U(z,k,t)) = C_{z+k-t-1}^{z-1} \cdot C_z^t \cdot {p_M}^{k-t} \cdot {p_H}^{z-t} \cdot {p_{HM}}^t.$$

 doi:10.20944/preprints202604.0694.v1

As the event $S(z) = k$ can be represented as a union of mutually exclusive events

$$\{S(z) = k\} = \bigcup_{t=0}^{\min\{z,k\}} U(z,k,t),$$

we can calculate $P_z(k)$ as

$$P_z(k) = \sum_{t=0}^{\min\{z,k\}} P(U(z,k,t)) = \sum_{t=0}^{\min\{z,k\}} C_{z+k-t-1}^{z-1} \cdot C_z^t \cdot {p_M}^{k-t} \cdot {p_H}^{z-t} \cdot {p_{HM}}^t,$$

and the Proposition is proved.

The next result we will formulate and prove is some kind of so-called "Gambler's Ruin Problem" [32], but adapted for our very model. It corresponds to the second phase of the double-spend Attack, which is typically referred to as the "catch-up" phase. The first phase of the attack lasts till the moment when the block with the initial transaction gets the set number of confirmation blocks (z blocks, in our designations). We assume that the vendor sends goods or services, paid in the initial transaction, just after he has seen the z-th confirmation block. If before this moment the malicious stakeholders managed to create a longer alternative chain (more than z blocks), they share this chain with the alternative transaction and get their payment back. If they managed to create only k blocks during the first phase, for some $0 \leq k < z$, they don't share it, but continue to create new blocks in this chain, hoping to "catch up" the longest chain created by honest stakeholders somewhere in the future. If it happens, they will share their longest chain and get back payment.

To calculate the probability of attack, we need to obtain the formula for the probability of event $Q_n = \{$malicious stakeholders will catch up the longest chain being n blocks behind$\}$.

Proposition 5. *In our designations,*

$$q_n = P(Q_n) = \begin{cases} \left(\frac{p_M}{p_H}\right)^n, \text{ if } p_M < p_H; \\ 1, \quad \text{else.} \end{cases}$$

Proof. Introduce events $E_H = \{$the next event is $H\bar{M}\}$, $E_M = \{$the next event is $\bar{H}M\}$, $E_{HM} = \{$the next event is $HM\}$, $C = \{$the next timeslot is empty$\}$. These events form a partition of sample space, so we can apply the Total Probability Law and write an equality:

$$Q_n = (Q_n \cap E_H) \cup (Q_n \cap E_M) \cup (Q_n \cap E_{HM}) \cup (Q_n \cap C),$$

and

$$q_n = P(Q_n/E_H) \cdot P(E_H) + P(Q_n/E_M) \cdot P(E_M) + P(Q_n/E_{HM}) \cdot P(E_{HM}) + P(Q_n/C) \cdot P(C),$$

which may be rewritten as

$$q_n = P(H\bar{M}) \cdot q_{n+1} + P(\bar{H}M) \cdot q_{n-1} + (P(HM) + 1 - f) \cdot q_n,$$

or

$$(f - P(HM))q_n = P(H\bar{M}) \cdot q_{n+1} + P(\bar{H}M) \cdot q_{n-1},$$

or

$$(P(H\bar{M}) + P(\bar{H}M))q_n = P(H\bar{M}) \cdot q_{n+1} + P(\bar{H}M) \cdot q_{n-1},$$

with boundary conditions $q_0 = 1$, $q_\infty = 0$ and $0 \leq q_n \leq 1$.

The corresponding characteristic equation is

$$(P(H\bar{M}) + P(\bar{H}M))\lambda^n = P(H\bar{M}) \cdot \lambda^{n+1} + P(\bar{H}M) \cdot \lambda^{n-1}$$

 doi:10.20944/preprints202604.0694.v1

or, as $\lambda \neq 0$,

$$P(H\bar{M}) \cdot \lambda^2 - (P(H\bar{M}) + P(\bar{H}M))\lambda + P(\bar{H}M) = 0. \quad (3)$$

We can rewrite (3) as

$$\lambda^2 - \left(1 + \frac{P(\bar{H}M)}{P(H\bar{M})}\right)\lambda + \frac{P(\bar{H}M)}{P(H\bar{M})} = 0, \quad (4)$$

from where $\lambda_1 = 1, \lambda_2 = \frac{P(\bar{H}M)}{P(H\bar{M})} = \frac{p_M}{p_H}$, where the last equality we get using (1) and (2).

If $p_M \geq p_H$, then $\lambda_2 = \frac{p_M}{p_H}$ can't be the solution because of the second boundary condition. So in this case $q_n = 1$ for $n \geq 1$.

If $p_M < p_H$, then $\lambda_2 = \frac{p_M}{p_H}$ is the only solution. In this case $q_n = \left(\frac{p_M}{p_H}\right)^n$ for $n \geq 1$.

The proposition is proved.

Note that in case $p_M = p_H$, the only solution of (4) is $\lambda = 1$, which means that the probability to catch up (during unlimited time) is equal to 1.

It should be noted that although the variant of the Gambler's Ruin Problem analyzed in Proposition 5 differs from the "classical" version, which has only two possible outcomes, the resulting probability of success follows the same form as in the classical case.

It is also interesting to examine some numerical results related to Proposition 5, which are presented below in Table 1. An adversary can use this proposition to evaluate the intermediate outcome of their attack. For instance, if they fall significantly behind in block production by the end of the first phase, they can calculate the probability of catching up and determine whether it is worthwhile to continue the attack.

Table 1 presents the probability of an adversary catching up in the future (over an unlimited time interval) when they are currently l blocks behind, for different adversary's ratios α_M.

Table 1. The probability to catch up in the second phase, being l blocks behind.

α_M	l				
	1	**5**	**10**	**20**	**30**
0.05	0.05263	4.039×10^{-7}	1.631×10^{-13}	2.660×10^{-26}	4.339×10^{-39}
0.1	0.11111	1.694×10^{-5}	2.868×10^{-10}	8.225×10^{-20}	2.359×10^{-29}
0.2	0.25000	0.00098	9.537×10^{-7}	9.095×10^{-13}	8.674×10^{-19}
0.3	0.42857	0.01446	0.0002	4.370×10^{-8}	9.135×10^{-12}

Now we are ready to formulate the main result of our paper about the probability of a double-spend attack for this model of PoS protocol.

Define P_z the probability of success of double-spend attack after z confirmation blocks (for given network parameters α_H, α_M, and active slot coeffecient f)

Proposition 6. *In our designations,*

$$P_z = 1 - \sum_{k=0}^{z-1} P_z(k)\left(1 - \left(\frac{p_M}{p_H}\right)^{z-k}\right).$$

Proof. According to Proposition 4, the probability that an attack isn't successful in the first phase is

$$P^* = \sum_{k=0}^{z-1} P_z(k) = \sum_{k=0}^{z-1}\sum_{t=0}^{k} P(U(z,k,t)) = \sum_{k=0}^{z-1}\sum_{t=0}^{k} C_{z+k-t-1}^{z-1} \cdot C_z^t \cdot {p_M}^{k-t} \cdot {p_H}^{z-t} \cdot {p_{HM}}^t,$$

and the probability that it succeeds in the first phase is $P_1 = 1 - P^*$.

Next, the event F that the attack fails on the first phase may be represented as a union of mutually exclusive events

 doi:10.20944/preprints202604.0694.v1

$$F = \bigcup_{k=0}^{z-1} F_k,$$

where $F_k = \{$malicious stakeholders create exactly k blocks on the first phase$\}$ and $P(F_k) = P_z(k)$.

Then the probability that, after failing in the first phase, the adversary catches up in the second one can be calculated as

$$P_2 = \sum_{k=0}^{z-1} P(Q_{z-k}/F_k) \cdot P(F_k) = \sum_{k=0}^{z-1} P_z(k) \cdot \left(\frac{p_M}{p_H}\right)^{z-k}.$$

Total probability P_z of successful attack is now

$$P_z = P_1 + P_2 = 1 - \sum_{k=0}^{z-1} P_z(k) + \sum_{k=0}^{z-1} P_z(k) \cdot \left(\frac{p_M}{p_H}\right)^{z-k} =$$
$$= 1 - \sum_{k=0}^{z-1} P_z(k) \left(1 - \left(\frac{p_M}{p_H}\right)^{z-k}\right),$$

and the proposition is proved.

4. Numerical Results

A Double-Spend Attack is one of the main attacks on the blockchain consensus protocol, and security against this attack is the pivotal point of the protocol's consistency. To prevent this attack, confirmation blocks are used. In practice, it is very important to figure out how many confirmation blocks may prevent this attack with preset overwhelming probability, like $1 - 10^{-3}$. In this paper, for the first time, we obtain an explicit formula for the attack probability, depending on network parameters and the number of confirmation blocks. Moreover, using this formula, it is easy to calculate the number of confirmation blocks sufficient to reduce the attack probability to any desired value. We give numerical results for different network parameters as examples of such calculations.

In this section, we present the numerical results obtained according to Proposition 6. Specifically, we chose several values of the active slot coefficient f and calculated the corresponding number of confirmation blocks we should wait to have the probability of a double-spend attack less than 10^{-3} (for different shares of malicious SLs). The results are presented in Table 2.

As we can see, in case of small adversary's ratio (like 0.05), the results are almost the same for different values of f. It means that the number of timeslots, obtained by stakeholder, may be considered proportionally to their stake ratio, which is a natural requirement for PoS consensus.

In the discussion below, $f = 0.05$ may be viewed as a Cardano-oriented reference setting already used in the literature on Ouroboros parameters, whereas larger values of f are included here as illustrative denser-slot regimes for sensitivity analysis rather than as one-to-one representations of specific deployed systems.

It is interesting to note that the higher the adversary's stake ratio, the greater the difference between the numerical values obtained for different values of the active slot coefficient f. As shown in Table 2, when the adversary's stake ratio α_M is 5% (or lower), four confirmation blocks are sufficient to ensure that the preceding blocks are finalized with a preset probability of 0.001. However, for a higher adversary's stake ratio, such as 40%, the required number of confirmation blocks increases from 133 to 149 as f increases (see the last row in Table 2).

Nevertheless, we cannot conclude that the absolute time required for block finalization necessarily increases with f, as it also depends on the length of timeslots and other network characteristics.

Table 2 is universal and can be used to determine block finalization requirements in different PoS-based blockchains with varying block creation intensities.

 doi:10.20944/preprints202604.0694.v1

Table 2. The number of confirmation blocks providing the probability of a double-spend attack less than 10^{-3} for different values of the active slot coefficient.

α_M	f					
	0.05	**0.1**	**0.2**	**0.5**	**0.8**	**0.9**
0.05	4	4	4	4	4	4
0.1	6	6	6	6	6	7
0.2	13	13	13	13	13	14
0.3	32	32	32	32	33	35
0.4	133	133	134	135	141	149

Table 3 shows some values P_z, according to Proposition 6, for a different ratio α_M of the adversary and different numbers of confirmation blocks z (for $f = 0.05$, the active slot coefficient is the same as in Cardano). A Vendor or customer may use these numerical results to define how many confirmation blocks they should wait to obtain the desired level of confidence (in terms of probability) that the block with their transaction is finalized. He may also find a desired trade-off between waiting time and the corresponding probability of attack on his block. For example, if the adversary's ratio is not larger than 35%, then 80-90 confirmation blocks guarantee that the probability of attack is almost negligible. For high-value transactions, it is recommended to increase the number of confirmation blocks to 120 or more.

Table 3. Double-spend attack probability for $f = 0.05$.

z	α_M				
	0.30	**0.33**	**0.35**	**0.40**	**0.45**
1	0.600037	0.660033	0.700030	0.800021	0.900011
2	0.432023	0.509674	0.563520	0.704015	0.850508
3	0.326180	0.409946	0.470357	0.634895	0.813754
4	0.252089	0.336446	0.399709	0.579599	0.783433
5	0.197632	0.279568	0.343455	0.533150	0.757167
10	0.065115	0.121586	0.174960	0.372199	0.657938
20	0.008676	0.027596	0.053580	0.204130	0.528642
30	0.001277	0.006858	0.017841	0.119114	0.439347
40	1.9675×10^{-4}	0.001777	0.006177	0.071629	0.371400
50	3.1110×10^{-5}	4.7197×10^{-4}	0.002189	0.043867	0.317318
60	5.0044×10^{-6}	1.2739×10^{-4}	7.8770×10^{-4}	0.027200	0.273129
70	8.1483×10^{-7}	3.4781×10^{-5}	2.8659×10^{-4}	0.017019	0.236396
80	1.3388×10^{-7}	9.5789×10^{-6}	1.0513×10^{-4}	0.010722	0.205485
90	2.2151×10^{-8}	2.6559×10^{-6}	3.8819×10^{-5}	0.006792	0.179236
100	3.6853×10^{-9}	7.4034×10^{-7}	1.4407×10^{-5}	0.004321	0.156787
110	6.1576×10^{-10}	2.0729×10^{-7}	5.3701×10^{-6}	0.002760	0.137480
120	1.0306×10^{-10}	5.8251×10^{-8}	2.0088×10^{-6}	0.001768	0.120799
130	1.7077×10^{-11}	1.6420×10^{-8}	7.5369×10^{-7}	0.001135	0.106331
140	2.5917×10^{-12}	4.6409×10^{-9}	2.8352×10^{-7}	0.000731	0.093741

Figure 2 gives the dependency of the logarithm value of probability P_z of a double-spend attack under z confirmation blocks (on the Y-axis) on the value z (on the X-axis), for different adversary ratios. As long as the graphics for the logarithm of probability are straight lines, then the value P_z decreases exponentially with the growth of z.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 10 April 2026 doi:10.20944/preprints202604.0694.v1

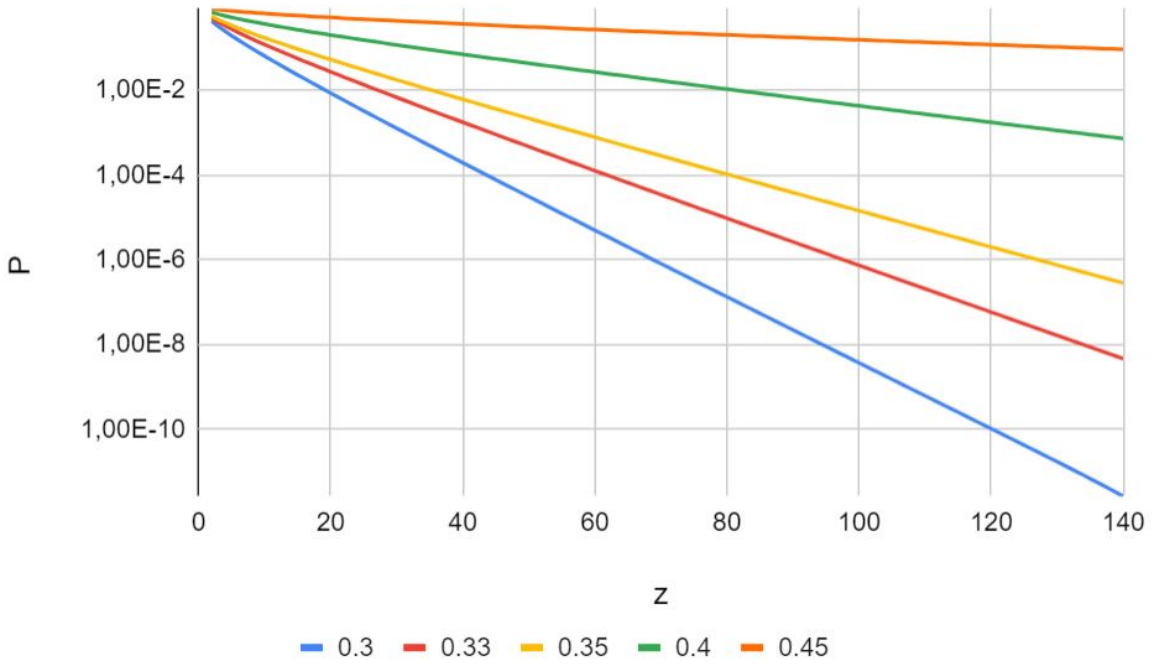

Figure 2. The logarithms of a double-spend attack probability for a different adversary ratio of α_M for $f = 0.05$.

To make the practical interpretation more explicit, Table 4 summarizes the minimum number of confirmation blocks required for several target risk thresholds in the reference setting $f = 0.05$. The values confirm the rapid growth of the required confirmation depth as the adversarial stake ratio approaches 0.4. In particular, the threshold 10^{-6} is reached already at 69 blocks for $\alpha_M = 0.30$, but requires 128 blocks for $\alpha_M = 0.35$ and 294 blocks for $\alpha_M = 0.40$.

Figure 3 visualizes the same tendency by showing how the minimum confirmation depth z increases with the adversarial stake ratio for three target risk levels. The figure confirms that the dependence is highly nonlinear: moderate increases in the adversarial ratio near 0.4 lead to a sharp increase in the required confirmation depth.

Table 4. Required confirmation depth z for different adversarial stake fractions α_M and attack probability thresholds ε.

α_M	$\varepsilon = 10^{-3}$	$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-6}$
0.30	32	44	69
0.33	45	62	98
0.35	58	81	128
0.40	133	186	294

Table 5 presents a validation of Proposition 6 by comparing the success probabilities of a double-spend attack computed using three independent methods: (i) the analytical formula derived in the manuscript, (ii) an independent dynamic programming (DP) implementation, and (iii) Monte Carlo simulations with 20,000 runs per case. The table includes several representative combinations of adversarial stake fraction α_M, active slot coefficient f, and confirmation depth z.

 doi:10.20944/preprints202604.0694.v1

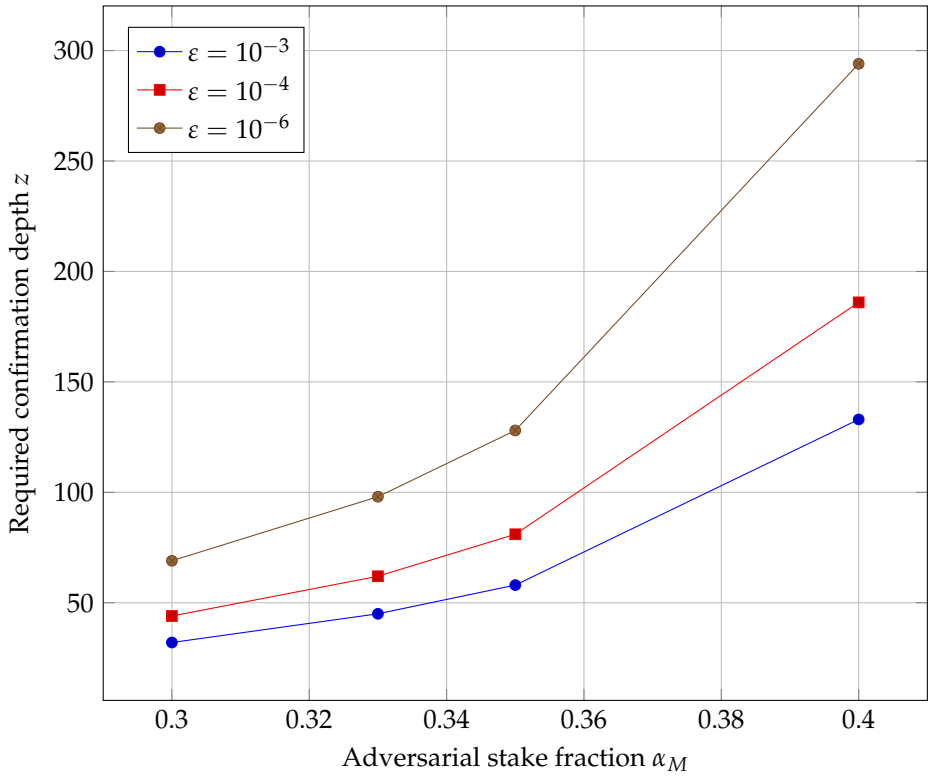

Figure 3. Confirmation depth z vs adversarial stake fraction α_M for different target attack probabilities ε.

The results show perfect agreement between the analytical formula and the DP check, while Monte Carlo estimates exhibit only minor deviations within the expected statistical error. This confirms the correctness of the analytical derivation and demonstrates that Proposition 6 accurately captures the attack probability within the adopted model assumptions.

Table 5. Validation of Proposition 6: comparison of analytical formula, independent DP check, and Monte Carlo simulations (20,000 runs).

α_M	f	z	P_z (Formula)	P_z (DP)	P_z (Monte Carlo)
0.10	0.05	2	0.056009	0.056009	0.057600
0.10	0.05	4	0.005457	0.005457	0.005650
0.20	0.20	4	0.066851	0.066851	0.066500
0.30	0.05	4	0.252089	0.252089	0.252350
0.30	0.50	4	0.255199	0.255199	0.254850
0.40	0.90	4	0.609586	0.609586	0.607350
0.40	0.90	8	0.456269	0.456269	0.460850

5. Discussion

The presented model incorporates multiple SLs and empty timeslots in the analysis of the PoS protocol security against double-spend attacks based on certain assumptions. So, we improved the previous results, bringing the model closer to real-world conditions within the adopted assumptions. We also obtained the corresponding numerical results on the probability of an attack, which confirmed the accuracy and utility of the analytical results.

The resulting formulas for calculating the probability of an attack allow not only the computation of this probability for any given set of network parameters but also the determination of the number of confirmation blocks corresponding to the desired probability. The numerical results obtained from these formulas once again confirm their accuracy and practical value.

An important practical observation follows from the numerical results presented in Tables 2 and 3. For relatively small adversarial stake ratios, the required number of confirmation blocks depends only weakly on the active slot coefficient f. However, as the adversarial stake ratio increases, this

 doi:10.20944/preprints202604.0694.v1

dependence becomes more noticeable. In particular, Table 2 shows that for low adversarial ratios the same confirmation depth is sufficient for a broad range of values of f, whereas for larger ratios the required depth gradually increases. This means that protocol parameters that may seem equivalent in a low-adversary regime can lead to visibly different security margins when the adversarial stake becomes closer to the critical threshold. Thus, the obtained formulas are useful not only for estimating attack probabilities, but also for comparing different operating regimes of the protocol.

The question of determining or estimating the total ratio of malicious stakeholders is interesting in itself, but falls outside the scope of this study. We simply note that this ratio can be estimated statistically by observing malicious activities in the network, such as the systematic artificial creation of forks or the frequent absence of a slot leader in its designated timeslot. However, we do not explore this issue in this paper.

More importantly, the key question is whether a double-spend attack can be prevented using confirmation blocks when the ratio of malicious stakeholders is below 50%. This is precisely what we prove in this paper within the adopted analytical model.

At the same time, the results also indicate a natural limitation of confirmation-based protection. When the adversarial stake remains below 50%, the attack probability decreases as the number of confirmation blocks grows, and this decrease is exponential in the numerical examples considered. However, the closer the adversarial ratio is to 50%, the slower this decrease becomes, and the larger the confirmation depth required in practice. Therefore, confirmation blocks are indeed an effective protection mechanism, but their efficiency is strongly regime-dependent: they work especially well when the adversarial stake is sufficiently separated from the critical threshold and become increasingly costly in terms of waiting time as this threshold is approached.

The main shortcoming of the model presented in this paper is a simplified assumption about a non-zero time delay. It may have an essential impact on attack probability only in the case when the time slot duration is essentially smaller than the time delay.

Our analysis is subject to several modeling assumptions. In particular, we do not consider multi-slot fork persistence, protocol-level finality rules (such as the Praos density rule), rollback bounds, inter-epoch dependencies, adversarial bias in randomness, or explicit latency-error bounds. These limitations define the scope of our results.

6. Conclusions

We present the model for a double-spend attack on PoS consensus with multiple SLs, i.e., where forks may occur whenever the corresponding timeslot has more than 1 slot leader. We use a notion of *active slot coefficient* f, introduced in [8], assuming that slots with an associated SL are active slots. We also assume that after each timeslot with at least 1 honest SL, the longest chain grows by 1 block, and the block issued in some timeslot is seen for all nodes at the beginning of the next timeslot.

In this work, we have extended previous results, refining the model to better reflect real-world conditions within the adopted assumptions. We have also computed corresponding numerical estimates for the attack probability, which validate both the accuracy and practical relevance of the analytical framework.

The model is completely scalable for an arbitrary number of stakeholders and an arbitrary stake distribution. This property is achieved due to the special properties of the block creation function $\phi_f(\alpha)$ (see Proposition 1).

Also, in the course of solving the problem, we obtained a generalization of the inverse binomial distribution (see Proposition 5), which is interesting from a mathematical point of view.

Another important conclusion concerns the interpretation of confirmation depth as a directly controllable security parameter. In asymptotic security analyses, one usually proves that security improves as the depth grows, but such statements do not immediately answer the practical question of how many confirmations are enough in a given network. In contrast, our results make it possible to pass from qualitative security guarantees to quantitative recommendations. In particular, a pro-

 doi:10.20944/preprints202604.0694.v1

tocol designer, vendor, or service provider can fix an acceptable attack probability in advance and then determine the minimal number of confirmation blocks required for the corresponding network parameters. This makes the obtained results directly applicable in settings where one must explicitly balance settlement latency and security.

Our results have significant practical implications for securing blockchain networks. They provide a clear method for calculating the number of block confirmations needed to make a transaction effectively irreversible. This allows protocol developers to fine-tune network parameters for resource-constrained devices, while service providers can programmatically decide how long to wait before an automated service is delivered, balancing security risks with operational speed. Ultimately, having such quantifiable security acts as a powerful deterrent, discouraging attacks on the network.

In our future work, we plan to generalize the model investigated here and incorporate a more complex assumption of non-zero time delay in the network. Previously, we had such an experience when we generalized the results obtained in [16] for the PoW consensus protocol in our work [17]. Such a generalization leads to more complex analytical proofs and estimations. We hope to achieve these results for the PoS protocol with multiple slot leaders and non-zero time delay in our future work.

Another natural direction for further research is to adapt the proposed approach to more general models of stake evolution and network behavior, including settings where the stake distribution changes across epochs or where synchronization assumptions are weakened. This would make it possible to extend the present explicit-probability framework to a broader class of Ouroboros-like protocols and deployment scenarios.

Author Contributions: Conceptualization, L.K.; methodology, L.K.; software, M.R.; validation, R.O. and V.A.; formal analysis, L.K.; investigation, L.K., M.R. and R.O.; resources, M.R.; data curation, M.R.; writing—original draft preparation, L.K. and M.R.; writing—review and editing, R.O. and V.A.; visualization, M.R.; supervision, R.O.; project administration, R.O.; funding acquisition, V.A. All authors have read and agreed to the published version of the manuscript.

Funding: The research was conducted as part of the project 'Development of Distributed Energy in the Context of the Ukrainian Electricity Market Using Digitalization Technologies and Systems,' implemented under the state budget program 'Support for Priority Scientific Research and Scientific-Technical (Experimental) Developments of National Importance' (CPCEL 6541230) at the National Academy of Sciences of Ukraine.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are available on request.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ASC	Active Slot Coefficient
BCF	Block Creation Function
DS	Double-spend
PoS	Proof-of-Stake
PoW	Proof-of-Work
SL	Slot Leader
TS	Timeslot

References

1. Sunny, F.A.; Hajek, P.; Munk, M.; Abedin, M.Z.; Satu, M.S.; Efat, M.I.A.; Islam, M.J. A systematic review of blockchain applications. *IEEE Access* **2022**, *10*, 59155–59177.

 doi:10.20944/preprints202604.0694.v1

2. Renduchintala, T.; Alfauri, H.; Yang, Z.; Pietro, R.D.; Jain, R. A survey of blockchain applications in the fintech sector. *Journal of Open Innovation: Technology, Market, and Complexity* **2022**, *8*, 185.
3. Al Hwaitat, A.K.; Almaiah, M.A.; Ali, A.; Al-Otaibi, S.; Shishakly, R.; Lutfi, A.; Alrawad, M. A New Blockchain-Based Authentication Framework for Secure IoT Networks. *Electronics* **2023**, *12*. https://doi.org/10.3390/electronics12173618.
4. AlSalem, T.S.; Almaiah, M.A.; Lutfi, A. Cybersecurity Risk Analysis in the IoT: A Systematic Review. *Electronics* **2023**, *12*. https://doi.org/10.3390/electronics12183958.
5. Nakamoto, S. A peer-to-peer electronic cash system **2008**.
6. Karpinski, M.; Kovalchuk, L.; Kochan, R.; Oliynykov, R.; Rodinko, M.; Wieclaw, L. Blockchain Technologies: Probability of Double-Spend Attack on a Proof-of-Stake Consensus. *Sensors* **2021**, *21*, 6408.
7. Kiayias, A.; Russell, A.; David, B.; Oliynykov, R. Ouroboros: A provably secure proof-of-stake blockchain protocol. In Proceedings of the Annual international cryptology conference. Springer, 2017, pp. 357–388.
8. David, B.; Gaži, P.; Kiayias, A.; Russell, A. Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain. In Proceedings of the Advances in Cryptology–EUROCRYPT 2018: 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29-May 3, 2018 Proceedings, Part II 37. Springer, 2018, pp. 66–98.
9. Badertscher, C.; Gaži, P.; Kiayias, A.; Russell, A.; Zikas, V. Ouroboros genesis: Composable proof-of-stake blockchains with dynamic availability. In Proceedings of the Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, 2018, pp. 913–930.
10. Badertscher, C.; Gazi, P.; Kiayias, A.; Russell, A.; Zikas, V. Ouroboros chronos: Permissionless clock synchronization via proof-of-stake. *Cryptology ePrint Archive* **2019**.
11. Ovezik, C.; Kiayias, A. Decentralization Analysis of Pooling Behavior in Cardano Proof of Stake. In Proceedings of the Proceedings of the Third ACM International Conference on AI in Finance, 2022, pp. 18–26.
12. Fitzi, M.; Gazi, P.; Kiayias, A.; Russell, A. Proof-of-stake blockchain protocols with near-optimal throughput, 2020.
13. Kiayias, A.; Koutsoupias, E.; Marmolejo-Cossío, F.; Stouka, A.P. Balancing Participation and Decentralization in Proof-of-Stake Cryptocurrencies. In Proceedings of the International Symposium on Algorithmic Game Theory. Springer, 2024, pp. 333–350.
14. Rosenfeld, M. Analysis of hashrate-based double spending. *arXiv preprint arXiv:1402.2009* **2014**.
15. Pinzón, C.; Rocha, C. Double-spend attack models with time advantange for bitcoin. *Electronic Notes in Theoretical Computer Science* **2016**, *329*, 79–103.
16. Grunspan, C.; Pérez-Marco, R. Double spend races. *International Journal of Theoretical and Applied Finance* **2018**, *21*, 1850053.
17. Kovalchuk, L.; Kaidalov, D.; Nastenko, A.; Rodinko, M.; Shevtsov, O.; Oliynykov, R. Decreasing security threshold against double spend attack in networks with slow synchronization. *Computer Communications* **2020**, *154*, 75–81.
18. Jang, J.; Lee, H.N. Profitable double-spending attacks. *Applied Sciences* **2020**, *10*, 8477.
19. Hafid, A.; Hafid, A.S.; Makrakis, D. Sharding-Based Proof-of-Stake Blockchain Protocols: Key Components & Probabilistic Security Analysis. *Sensors* **2023**, *23*, 2819.
20. Li, A.; Wei, X.; He, Z. Robust proof of stake: A new consensus protocol for sustainable blockchain systems. *Sustainability* **2020**, *12*, 2824.
21. Naz, S.; Lee, S.U.J. Sea Shield: A Blockchain Technology Consensus to Improve Proof-of-Stake-Based Consensus Blockchain Safety. *Mathematics* **2024**, *12*, 833.
22. Mao, H.; Nie, T.; Yu, M.; Dong, X.; Li, X.; Yu, G. SMPTC3: Secure Multi-Party Protocol Based Trusted Cross-Chain Contracts. *Mathematics* **2024**, *12*, 2562.
23. Gao, Y.; Li, X.; Peng, Z.; Zhang, Y.; Yu, G. NeuChain+: A Sharding Permissioned Blockchain System with Ordering-Free Consensus. *Applied Sciences* **2024**, *14*, 4897.
24. Chegenizadeh, M.; Li, S.N.; Tessone, C.J. Towards a deeper understanding of the Cardano macro-economics. In Proceedings of the 2024 IEEE International Conference on Blockchain and Cryptocurrency (ICBC). IEEE, 2024, pp. 585–593.
25. Alshahrani, H.; Islam, N.; Syed, D.; Sulaiman, A.; Al Reshan, M.S.; Rajab, K.; Shaikh, A.; Shuja-Uddin, J.; Soomro, A. Sustainability in blockchain: A systematic literature review on scalability and power consumption issues. *Energies* **2023**, *16*, 1510.

 doi:10.20944/preprints202604.0694.v1

26. Cao, B.; Zhang, Z.; Feng, D.; Zhang, S.; Zhang, L.; Peng, M.; Li, Y. Performance analysis and comparison of PoW, PoS and DAG based blockchains. *Digital Communications and Networks* **2020**, *6*, 480–485.
27. Fahim, S.; Rahman, S.K.; Mahmood, S. Blockchain: A comparative study of consensus algorithms PoW, PoS, PoA, PoV. *Int. J. Math. Sci. Comput* **2023**, *3*, 46–57.
28. Lepore, C.; Ceria, M.; Visconti, A.; Rao, U.P.; Shah, K.A.; Zanolini, L. A survey on blockchain consensus with a performance comparison of PoW, PoS and pure PoS. *Mathematics* **2020**, *8*, 1782.
29. Rukhiran, M.; Boonsong, S.; Netinant, P. Sustainable Optimizing Performance and Energy Efficiency in Proof of Work Blockchain: A Multilinear Regression Approach. *Sustainability* **2024**, *16*, 1519.
30. Hossan, M.R.; Nirob, F.A.; Islam, A.; Rakin, T.M.; Al-Amin, M. A Comprehensive Analysis of Blockchain Technology and Consensus Protocols Across Multilayered Framework. *IEEE Access* **2024**.
31. Ahn, J.; Yi, E.; Kim, M. Blockchain consensus mechanisms: A bibliometric analysis (2014–2024) using vosviewer and r bibliometrix. *Information* **2024**, *15*, 644.
32. Feller, W. *An introduction to probability theory and its applications, Volume 2*; Vol. 81, John Wiley & Sons, 1991.