TU Delft

Delft University of Technology

Leveraging the Verifier's Dilemma to Double Spend in Bitcoin

Cao, Tong; Decouchant, J.E.A.P.; Yu, Jiangshan

DOI
10.1007/978-3-031-47751-5_9

Publication date
2023

Document Version
Final published version

Published in
Financial Cryptography and Data Security

Citation (APA)
Cao, T., Decouchant, J. E. A. P., & Yu, J. (2023). Leveraging the Verifier's Dilemma to Double Spend in Bitcoin. In F. Baldimtsi, & C. Cachin (Eds.), *Financial Cryptography and Data Security* (pp. 149-165). (Lecture Notes in Computer Science). https://doi.org/10.1007/978-3-031-47751-5_9

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Green Open Access added to TU Delft Institutional Repository

'You share, we take care!' - Taverne project

https://www.openaccess.nl/en/you-share-we-take-care

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

Leveraging the Verifier's Dilemma to Double Spend in Bitcoin

Tong Cao[1(✉)], Jérémie Decouchant[2], and Jiangshan Yu[3]

[1] Kunyao Academy, Shanghai, China
tongcaodaniel@gmail.com
[2] Delft University of Technology, Delft, The Netherlands
[3] Monash University, Melbourne, Australia

Abstract. We describe and analyze *perishing mining*, a novel block-withholding mining strategy that lures profit-driven miners away from doing useful work on the public chain by releasing block headers from a privately maintained chain. We then introduce the *dual private chain (DPC) attack*, where an adversary that aims at double spending increases its success rate by intermittently dedicating part of its hash power to *perishing mining*. We detail the DPC attack's Markov decision process, evaluate its double spending success rate using Monte Carlo simulations. We show that the DPC attack lowers Bitcoin's security bound in the presence of profit-driven miners that do not wait to validate the transactions of a block before mining on it.

Keywords: Bitcoin · Double spending · Block withholding attack

1 Introduction

Bitcoin's security level is traditionally measured as the proportion of the mining power that an adversary must control to successfully attack it. Nakamoto assumed that an adversary would not control the majority of the mining power [28]. If this assumption does not hold, an attacker is able to spend a coin twice and affect the system consistency in what is known as a double spending attack or 51% attack. The soundness of the honest majority assumption has been discussed in the literature and mechanisms have been proposed to harden the mining process against the 51% attack without completely eliminating it [8,10,23,37].

Despite rewarding miners with newly minted coins and transaction fees, the Bitcoin mining process has also been shown to be vulnerable to selfish behaviors. Using selfish mining, a miner withholds mined blocks and releases them only after

This work was partly performed while Tong Cao was with the University of Luxembourg.

J. Decouchant and J. Yu—These authors are listed in alphabetical order and contributed equally.

© International Financial Cryptography Association 2024
F. Baldimtsi and C. Cachin (Eds.): FC 2023, LNCS 13951, pp. 149–165, 2024.
https://doi.org/10.1007/978-3-031-47751-5_9

the honest miners have wasted computing resources mining alternative blocks. Selfish mining increases a miner's revenue beyond the fair share it would obtain by following the default Bitcoin mining protocol [19]. Using simulations, selfish mining has been shown to be profitable only after a difficulty adjustment period in Bitcoin for any miner with more than 33% of the global hash power [21,30]. Variants of selfish mining further optimize a miner's expected revenue [34].

Additionally, miners face the verifier's dilemma [7,26,36], where upon receiving a block header they have to decide whether they should wait to have received and verified the corresponding transactions, or whether they should start mining right away based on the block header. Different miners might react differently to this dilemma.

Following previous works, we say that a chain of blocks is public if the honest miners are able to receive all its content, while we say that a chain is private if some contents of the chain are kept hidden by the adversary. In this paper, we show that an adversary can leverage a novel block withholding strategy, which we call perishing mining, to slow down the public chain in an unprecedented manner. More precisely, perishing mining leads miners that react differently to the verifier's dilemma to mine on different forks. We then present the Dual Private Chain (DPC) attack, which further leverages the verifier's dilemma to double spend on Bitcoin. This attack is, to the best of our knowledge, the first attack where an adversary temporarily sacrifices part of its hash power to later favor its double spending attack, and the first attack where an adversary simultaneously manages two private chains. Intuitively, the first adversarial chain inhibits the public chain's growth, so that the second one benefits from more favorable conditions for a double spending attack.

To evaluate the impact of the distraction chain on the public chain we first establish the Markov decision process (MDP) of perishing mining. From this MDP, we obtain the probability for the system to be in each state, and quantify the impact of perishing mining on the public chain, i.e., its growth rate decrease. We further describe the DPC attack and its associated MDP. We then evaluate its expected success rate based on Monte Carlo simulations. Counterintuitively, our results show that the adversary increases its double spending success rate by dedicating a fraction of its hash power to slow the public chain down, instead of attacking it frontally with all its hash power.

Overall, this work makes the following contributions.

- We present perishing mining, a mining strategy that is tailored to slow down the progress of the public chain by leveraging the verifier's dilemma. Using perishing mining an adversary releases the headers of blocks that extend the public chain so that some honest miners mine on them while some honest miners keep mining on the public chain, which effectively divides the honest miners hash power. We present the pseudocode of the perishing mining strategy, establish its Markov chain model and quantify its impact on the public chain growth.
- Building on perishing mining, we describe the DPC attack that an adversary can employ to double spend by maintaining up to two private chains. The first chain leverages the perishing mining strategy to slow down the public chain's

growth and ease the task of the second chain, which aims at double spending. We provide the pseudocode of the attack, and characterize the states and transitions of its Markov chain model.

• We evaluate the perishing mining strategy and the DPC attack based on extensive Monte Carlo simulations. Our results indicate that perishing mining reduces the public chain progress by 69% when the adversary owns 20% of the global power and 50% of the hash power belongs to miners that mine on block headers without verifying their transactions. In comparison, selfish mining, which aims at optimizing a miner's revenue share, would only decrease it down by 15%. Our evaluation also shows that an adversary that owns 30% of the global hash power can double spend with 100% success rate when 50% of the hash power belongs to optimistic miners who do not verify transactions (i.e., type 2 miners in Sect. 3.2). While we focus on the double spending threat, we also show that the DPC attack allows an adversary to obtain a higher revenue than the one it would obtain by mining honestly or following previously known strategies (Appx. ??).

This paper is organized as follows. Section 2 discusses the related work and provides some necessary background. Section 3 defines our system model. Section 4 provides an overview of the DPC attack. Section 5 details the perishing mining strategy and the DPC attack that builds on it. Section 6 presents our evaluation results. Section 7 provides a discussion on other aspects of the attack. Finally, Sect. 8 concludes this paper.

2 Related Work

Double Spending Attack. The double spending attack on Bitcoin was described in Nakamoto's whitepaper [28], and has been further analyzed since [25,33]. Nowadays, $z = 6$ blocks need to be appended after a block for its transactions to be considered permanent. An adversary with more than 50% of the global mining power is able to use a coin in a first validated transaction and, later on, in a second conflicting transaction. Nakamoto characterized the race between the attacker and the honest miners as a random walk, and calculated the probability for an attacker to catch up with the public chain after z blocks have been appended after its initial transaction. Our DPC attack aims at double spending, and improves upon the classical double spending's success rate.

Block-Withholding Attacks. Selfish mining was the first mining strategy that allows a rational miner to increase its revenue share [19], and was later shown to harm the mining fairness [9,15]. Selfish mining is not more profitable than honest mining when the mining difficulty remains constant despite the fact that the adversary is able to increase its revenue share [21,22]. Nayak et al. proposed plausible values for the selfish miner's propagation factor by utilizing the public overlay network data [29]. They pointed out that the attacker could optimize its revenue and win more blocks by eclipsing [24] honest miners when the propagation factor increases. Gervais et al. analyzed the impact of stale rate on selfish mining attack [21]. Negy et al. pointed out that applying selfish mining

in Bitcoin is profitable after at least one difficulty adjustment period (i.e., after approximately two weeks at least) [30]. The DPC attack differs from these works in the sense that its main goal is not to increase the adversary's mining share but to double spend with higher probability than previous attacks.

Table 1. Notations.

Symbol	Interpretation
$\alpha \in [0, 0.5]$	Mining power of the adversary
$\beta \in [0, 1]$	Fraction of its mining power that the adversary dedicates to its first private chain
$\mu \in [0, 0.5]$	Mining power of type 2 miners
v_t	Value of the transaction the adversary inserts in a block when starting the DPC attack and attempts to double spend
v_b	Mining reward per block

Combining Selfish Mining and Double Spending. Previous works have shown that an adversary can combine the double spending attack with selfish mining [21,35]. In this attack, the attacker maintains a single chain, which lowers the double spending success rate compared to the initial double spending attack. Our DPC attack shows that an adversary can simultaneously manage two private chains to launch a more powerful double spending attack.

Blockchain Denial of Service Attacks. The BDOS attack proposed strategies to partially or completely shut down the mining network [27]. To do so, the adversary only sends the block header to the network whenever she discovers a block that is ahead of the public chain and there is no fork, and publishes the block body if the next block is generated by the honest miners. By doing so, the profitability and utility of the rational miners and Simplified Payment Verification (SPV) miners is decreased, so that they eventually leave the mining network. The objective of BDOS attacks is to halt the system. An adversary would need to spend approximately 1 million USD per day to shut down the system. Our DPC attack frequently separates other miners' hash power, which has some similarities with the BDOS attack's partial shut down case. However, the DPC attack allows the adversary to double spend.

3 System Model

This section introduces the categories of miners we consider, and the adversary that launches a DPC attack. Table 1 summarizes our notations.

3.1 Bitcoin Mining and the Verifier's Dilemma

Bitcoin mining is a trial-and-error process[1]. The public blockchain (or chain) is visible to all participants, and is maintained by honest miners. To achieve

[1] https://en.bitcoin.it/wiki/Block_hashing_algorithm.

consistency, honest miners accept the longest chain in case of visible forks [17,20,31]. However, temporary block withholding attacks have been shown to threaten Bitcoin's security [19,25,33,34]. Honest miners monitor the network to verify block headers and verify transactions.

In the Bitcoin's network, block headers are often propagated faster than transactions. Bitcoin's incentive mechanism does not directly reward the verification of transactions, and BIP-152[2] introduced the compact block propagation optimization where each node can relay a block in a compact format before verifying its transactions. In this case, a miner that immediately mines on the block header of a correct block gets a time advantage to find the next block. If the miners instead wait and verify the included transactions before the next mining round, then they might sacrifice some non-negligible time in the mining race [7,12,26,36].

We assume that miners follow the traditional block exchange pattern [16,27] using the overlay network. Block dissemination over the overlay network takes seconds, whereas the average mining interval is 10 min. While accidental forks (which may occur every 60 blocks [16] on average) reduce the effective honest mining power on the public chain and makes our attack easier, we do not consider accidental forks created by honest miners in order for simplicity. We evaluate mining and double spending strategies using event-based simulations where an event is the discovery of a block by a category of miner. We note v_b the mining reward that miners obtain whenever a block they have discovered is permanently included in the blockchain.

3.2 Miner Categories

We consider two types of honest Bitcoin miners that react differently to the verifier's dilemma: *type 1 honest miners* and *type 2 honest miners*.

Type 1 honest miners always follow the default mining protocol and mine on the longest chain of fully verified blocks. In particular, these miners do not mine on a block header that extends a longer non-fully verified concurrent chain.

Type 2 honest miners are profit-driven. As Bitcoin allows miners to accept and generate new blocks without verifying their transactions, type 2 miners start mining on a new block or its header if it extends the longest chain without verifying the transactions it contains. Note that type 2 miners can verify transactions whenever they are received and stop mining on a block header when associated transactions are faulty, or if they successfully mine the next block without having received the previous transactions. In our experiments, we consider two opposite categories of type 2 miners that behave differently upon reception of successive block headers to evaluate the best and worst possible attack results.

– *Optimistic type 2 miners* miners always mine on the longest chain of blocks, which is possibly made of several block whose transactions have not yet been received. In particular, Simplified Payment Verification (SPV) miners [3–6] can

[2] https://github.com/bitcoin/bips/blob/master/bip-0152.mediawiki.

be categorized as optimistic type 2 miners. Upon finding a block, optimistic type 2 miners can create an empty block or include transactions that they know cannot create conflicts (e.g., internal transactions for mining pools).
– *Pessimistic type 2 miners* only accept to mine on a block header if it extends a chain of full blocks. In particular, a pessimistic type 2 miner that extends a block header would then mine on the last block with transactions not to waste time. If the missing transactions eventually arrive, they then release the next full block. While if they extend over the last full block, they then create a fork.

In practice, it would be difficult for the adversary to identify the exact proportion of the global mining power that each type 2 miner subcategory represents. However, the adversary can be conservative and assume that all type 2 miners are pessimistic, since our attack still improves over the state-of-the-art in that case. We also discuss evidence for SPV mining in Sect. 7, which is arguably the simplest type 2 mining strategy.

The adversary owns a fraction $\alpha \in [0, 0.5]$ of the global hash power and its aim is to double spend with higher probability than using previous attacks. When launching its attack the adversary introduces a transaction of value v_t in a block that is included in the public chain and that it attempts to double spend. We also assume that the adversary cannot break cryptographic primitives. Contrary to the selfish mining's adversary model [19,21], our model does not assume that the adversary has a privileged network access, which is required in selfish mining when the adversary releases a conflicting block it had pre-mined in reaction to the extension of the public chain by an honest miner. For simplicity, we consider that every newly discovered and propagated block is almost instantaneously received by all miners. Several works evaluated and modeled network propagation delays in various cryptocurrencies [12,13,16].

4 Attack Overview

This section provides a high-level description of the Dual Private Chain (DPC) attack, where an adversary maintains two private chains. It then summarizes the respective roles of adversary's two private chains and their interactions.

4.1 Intuition

In a DPC attack, the adversary maintains two private chains from which it might release block headers or full blocks with the ultimate goal of double spending. During the attack, both of the adversary's private chains compete with the public chain and may diverge from it starting from different blocks. At a given point in time, the adversary might dedicate its full hash power to one of its private chains, or divide its hash power to simultaneously extend both private chains.

The DPC attack starts when the adversary creates a transaction of value v_t that is the basis for its double spending attempt. Once the adversary generates

the block that contains this transaction, she initializes both its private chains with it and starts mining on it. Initially, the two chains are therefore equal, but they might diverge or converge again later on depending on the created blocks. The double spending attack succeeds if the double spending chain becomes longer than the public chain and if the public chain contains $z = 6$ blocks that have been included after the block that contains the initial transaction of the adversary.

Role of the Distraction Chain. The first private chain that the adversary maintains is called the *distraction chain*. We present perishing mining, a strategy that the adversary employs to maintain its first private chain to waste the hash power of type 2 honest miners and slow down the public chain. Whenever the adversary divides its hash power to simultaneously mine on its two private chains, it dedicates β of its hash power to mine on its first private chain. This chain is private in the sense that the adversary never releases the full blocks, but only the corresponding block headers. The strategy that the adversary applies on its distraction chain divides the honest miners so that they mine on different blocks, and wastes the hash power of type 2 honest miners, which collectively account for hash power μ. The adversary leverages a BDOS-like attack to only share the header of blocks it discovers on the distraction chain (see Sect. 5). As the body of those blocks contain adversary-created transactions that are never publicly released, only type 2 honest miners mine on them. In this way, the adversary can distract type 2 honest miners from mining on the public chain.

Role of the Double Spending Chain. The adversary maintains a second private chain to attempt to double spend, and we therefore call this chain the *double spending chain*. Whenever the adversary is simultaneously mining on its two private chains it dedicates $\alpha(1-\beta)$ of the global hash power to its second private chain. This chain is private in the sense that, even though block headers might be released, the actual blocks it contains are only published if the double spending attack is successful. Following previous analyses [28,33], we consider that a double spending attempt is successful when: (i) the double spending chain's length is larger than or equal to the public chain's length; and (ii) z-1 blocks have been appended after the block that contains the adversary's initial transaction ($z = 6$ in Bitcoin).

4.2 Interplay Between the Two Private Chains

Whenever type 1 and type 2 miners are mining on the same block, the adversary divides its hash power to concurrently mine with hash power $\alpha\beta$ on the last block of its distraction chain, which is then equal to the public chain, and mine with mining power $\alpha(1 - \beta)$ on its double spending chain. The adversary's goal is then to create a fork and release a block header so that type 1 and type 2 honest miners mine on different blocks. Note that the adversary will use all its hash power on the second private chain as long as the first private chain is longer than the public chain. This hash power shifting between two private chains is at the core of the DPC attack, which is detailed in Sect. 5.2.

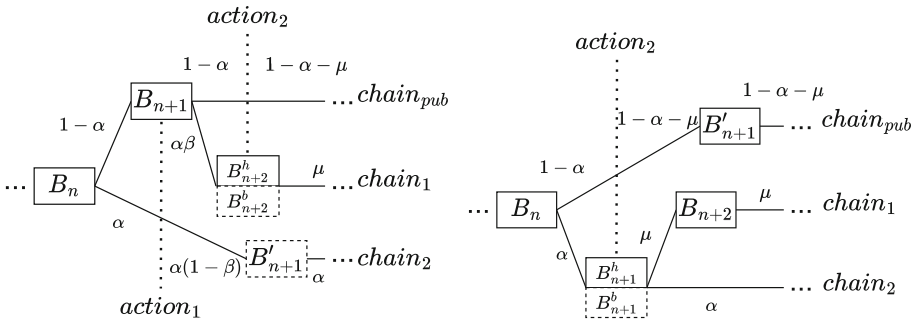


Fig. 1. Illustration of two possible cases that would lead type 2 miners to waste their hash power during a DPC attack. $B_n, B_{n+1}, B'_{n+1}, B_{n+2}$ are full blocks, B^h_{n+1}, B^h_{n+2} are block headers, and B^b_{n+1}, B^b_{n+2} are block bodies. We use solid rectangles when the content of a block is visible to honest miners, and a dotted rectangle when it is hidden by the adversary. We note interesting adversary's actions with $action_1$ and $action_2$ (see text for explanations).

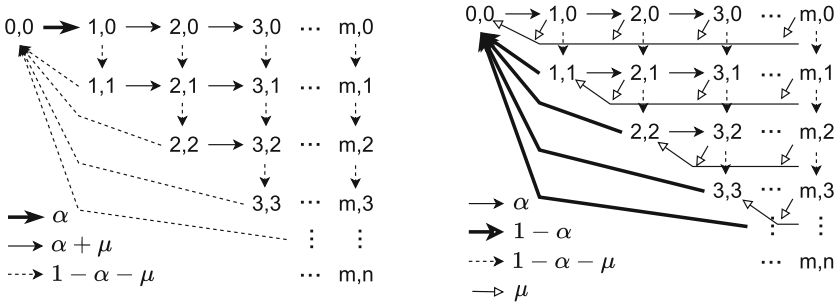
In the DPC attack, the adversary executes different actions to lead the honest miners to mine on different blocks. Figure 1 shows two possible scenarios where the attack is initialized based on block B_n. The adversary generates a pair of conflicting transactions for its double spending attack. The first transaction is released to the public network and collected by the honest miners. The second transaction is kept private by the adversary. In both examples, after $action_1$, the adversary separates her hash power into two parts: she uses $\alpha\beta$ to work on public block lead B_{n+1}, and $\alpha(1-\beta)$ to work on extending $chain_2$ to double spend. After $action_2$ the adversary releases the block header and uses all of her hash power to extend $chain_2$ for double spending. In both cases, type 2 honest miners (with μ of global hash power) are led to generate some blocks that will never be included in the public chain due to the adversary's block body withholding strategy. Consequently, the adversary's second private chain $chain_2$, which is used to attempt to double spend, benefits from the distraction of $chain_1$. We detail the DPC attack in Sect. 5.

5 The Dual Private Chains Attack

This section presents the details of the DPC attack, which attempts to lure type 2 honest miners away from extending the public chain, thus, facilitates a double spending attack. We first describe perishing mining, a strategy that a miner can use to slow down the progress of the public chain by making honest miners mine on different blocks. We then describe the full DPC attack that builds on perishing mining to maintain the adversary's first private chain. We provide an additional discussion on the DPC attack in Sect. 7.

5.1 Perishing Mining

We call *perishing mining* the strategy that the adversary uses on the distraction chain (whenever she is mining on it). After the initialization of the perishing mining strategy, the distraction chain and the public chain mine on the same block. The adversary's action then depends on whether the next block is discovered by the public miners or by itself (Please see our original analysis for details [11]). First, when the adversary discovers a block B_{n+1} that makes its distraction chain longer than the public chain, it releases the corresponding block header to the network. Upon receiving this header, type 2 miners start mining based on it, while type 1 miners continue working on block B_n. Second, when type 1 miners discover a block, the public chain is extended. Third, when type 2 miners find a block, the public chain is extended when the public chain is equal to the private chain. Otherwise, the block is abandoned due to the incomplete block verification, which wastes the hash power of type 2 miners. Note that when type 2 miners are optimistic the private chain is extended when it is not equal to the public chain.



(a) With optimistic type 2 miners.

(b) With pessimistic type 2 miners.

Fig. 2. Perishing mining's Markov chain models with optimistic and pessimistic type 2 miners. Arrows that do not lead to a state (on the right subfigure) represent the wasted mining effort of pessimistic type 2 miners. Only the top-left transition on the left figure has probability α.

Figure 2 illustrates the MDP models of the perishing mining strategy assuming that type 2 miners are either optimistic or pessimistic. In this Markov chains, α, μ and $1-\alpha-\mu$, are respectively the probabilities for the adversary, type 2 and type 1 miners to discover a block. We use a tuple (i, j) to denote the state in perishing mining's MDP, where i and j are respectively the lengths of the private chain and the public chain. The fact that the adversary adopts the public chain whenever it is longer than the private chain implies that $i \leq j$. The adversary releases the header of the leading block to lure type 2 miners. When type 2 miners are optimistic (Fig. 2a), the adversary relies on type 2 miners that also attempt to extend the private chain. When type 2 miners are pessimistic

(Fig. 2b), the adversary is not able to use them to extend the private chain. We evaluate the negative impact of perishing mining on the public chain growth in Sect. 6.2.

5.2 Combining Perishing Mining and Double Spending

The DPC attack leverages the perishing mining strategy to distract type 2 miners and facilitate double spending.

We detail the attack's pseudocode in our original analysis [11], where l_1, l_2, and l_{pub} represent the length of the first private chain $chain_1$, the second private chain $chain_2$, and the public chain $chain_{pub}$ respectively.

During the DPC attack, the two invariants $l_2 \leq l_1$ and $l_{pub} \leq l_1$ are verified. The distraction chain is therefore always the longest chain among the three chains, and can adopt the public chain and the double spending chain when it is not the longest chain. For example, if it happens that the double spending becomes the longest chain then the distraction chain is set to be equal to the double spending chain. As a consequence, the type 2 miners would mine on the headers of the double spending chain, which would facilitate the double spending attack.

When the DPC attack starts, all three chains are equal and all miners mine on the same block. The adversary's actions are defined in reaction to block discoveries.

When the adversary finds a block on the distraction chain, it releases the corresponding block header so that type 2 miners mine on it, because the distraction chain is then the longest chain. If the two private chains are equal, the newly found block also extends the double spending chain. As a consequence, the adversary extends the distraction chain, and type 1 miners mine on the last full block of the public chain while type 2 miners mine on the last block header of the distraction chain. The adversary then allocates all its hash power (α) to mining on the double spending chain.

When the adversary finds a block on its double spending chain, it releases the block header if the second private chain becomes the longest chain. In this case, type 2 miners then mine on the double spending chain. The first private chain also adopts the second private chain so that the total hash power used to extend the double spending chain is $\alpha + \mu$. When the second private chain is shorter than the public chain, the adversary keeps mining on it with $1 - \beta$ of its hash power. As soon as the double spending chain becomes longer than the public chain and that at least 6 blocks have been appended to the public chain since the beginning of the attack, the adversary uses the double spending chain to override the public chain, and the DPC attack succeeds.

When type 1 miners find a block, they extend the public chain. If the public chain becomes the longest chain, then all honest miners will mine on the public chain and the adversary modifies its first private chain so that it adopts the public chain. The adversary then allocates $\alpha\beta$ of hash power to its distraction chain so that it tries to generate a block that will divide again the honest miners.

When type 2 miners find a block, three cases are possible. First, the double spending chain is extended if two private chains are equal and longer than the public chain. Second, the public chain and first private chain are extended if they are equal. Finally, in the other cases the newly discovered block is abandoned, which wastes the hash power of type 2 honest miners. The DPC attack can be tailored to optimistic or pessimistic type 2 miners.

5.3 Markov Decision Process of the DPC Attack

We establish the Markov decision process (MDP) of the DPC attack by simultaneously considering the two private chains and observing that each state is a 5-tuple $(l_{pub}, l_1, l_2, s_{(pub,1)}, s_{(1,2)})$. l_{pub}, l_1, and l_2 are respectively the lengths of the public chain $chain_{pub}$, the first private chain $chain_1$, and the second private chain $chain_2$. $s_{(pub,1)}, s_{(1,2)} \in \{\texttt{t(rue)}, \texttt{f(alse)}\}$ respectively indicate whether $chain_{pub}$ is equal to $chain_1$, and whether $chain_1$ is equal to $chain_2$.

Based on the relations between the three chains (synchronized or not), we identified 10 types of states in the presence of optimistic type 2 miners, and 9 types of states in presence of pessimistic type 2 miners. The corresponding transitions are presented in our original analysis [11]. Note that we were not able to obtain closed form formulas for the probabilities of each possible state due to the complexity of the DPC attack's MDP model. Nevertheless, we use Monte Carlo simulations to estimate the adversary's success rate and revenue, as in previous block withholding attacks [19,21,29].

Case 0 is the initial state of the attack. Case 4 captures the attack success, which happens if the public and the double spending chains contain more than 6 blocks, and if the double spending chain is longer than the public chain. Cases 1.x, 2.x, 3.x are all possible intermediary states and consider scenarios that differ based on the lengths of the chains, and whether or not they are equal, which happens when the adversary reinitializes one or both of its private chains.

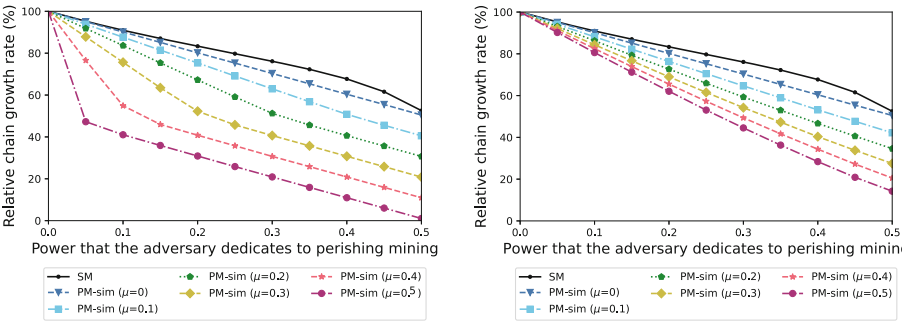
We emphasize that an adversary that executes the DPC attack earns a mining reward only when the double spending chain succeeds. In this case, the adversary earns the block mining reward that corresponds to the private blocks it mined that end up in the public chain and the value of the transaction it managed to double spend. We use v_b for the value of blocks, and v_t for the value of the double spent transactions.

6 Analysis Using Monte Carlo Simulations

This section evaluates the perishing mining strategy and the DPC attack using Monte Carlo simulations that react based on the event of block discovery.

6.1 Methodology and Settings

We evaluate perishing mining and the DPC attack using random walks in their respective Markov decision processes. Our evaluations are based on Python



(a) With optimistic type 2 miners. (b) With pessimistic type 2 miners.

Fig. 3. Relative growth rate of the public chain (compared to the attack-free case) when the adversary uses selfish mining (SM) or perishing mining (PM), where type 2 miners own a fraction μ of the global power.

scripts. In each scenario, we simulate the creation of 2,016 blocks, repeat each configuration 10,000 times, and report the average of the metrics of interest. Simulating the creation of 2,016 blocks maintains the mining difficulty constant during the experiment since Bitcoin's mining difficulty is adjusted every 2,016 blocks. We quantify the impact of perishing mining on the public chain's growth rate, and then evaluate the double spending success rate of the DPC attack. We compare the success rate of the DPC attack to the success rate of the classical double spending attack using the success rate formulas that were obtained by Nakamoto [28] and Rosenfeld [32]. We study the various strategies with $\alpha, \mu \in [0, 0.1, 0.2, 0.3, 0.4, 0.5]$ and $\beta \in [0, 1]$ (by 0.01 steps). Moreover, we analyze the adversary's expected revenue in our original analysis [11].

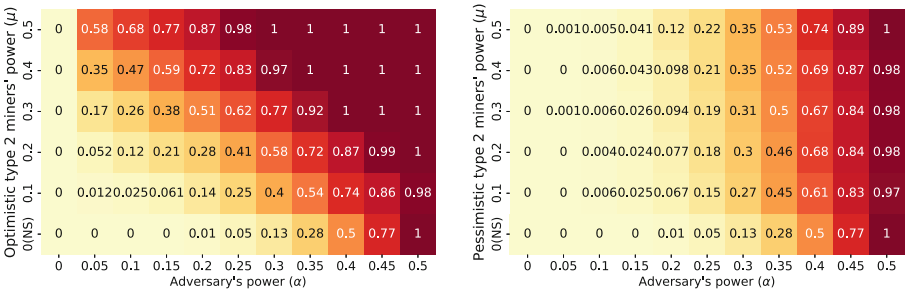
6.2 Impact of Perishing Mining on Chain Growth

In a DPC attack, the adversary leverages perishing mining strategy to inhibit public chain's growth. We now consider a scenario where the adversary constantly dedicates a fraction of its full hash power to perishing mining, so that we can quantify its effect on the growth rate of the public chain.

Figure 3 represents the relative public chain growth rate of a system under attack, which is expressed as a fraction (in %) of the public chain growth rate in the attack-free case. We compare perishing mining to selfish mining and vary the global hash power μ of type 2 miners 0 to 0.5 (i.e., ranging from 0% to 50% of the global hash power). The public chain is extended at a lower rate when the adversary's power increases and when the global power of type 2 miners increases. By comparing Fig. 3a and Fig. 3b, one can see that perishing mining has a stronger impact with optimistic type 2 miners than with pessimistic type 2 miners, as one could expect.

6.3 Double Spending Success Rate

Figure 4 illustrates the success rates of the DPC attack for different μ and with the best β value that we obtained experimentally. It is interesting to observe the differences between the partitions corresponding to a given μ with the best β value to see that maintaining distraction chain and double spending chain simultaneously makes a real difference. An adversary would be able to determine the best β after estimating μ, as we discuss in Sect. 7.



(a) With optimistic type 2 miners.

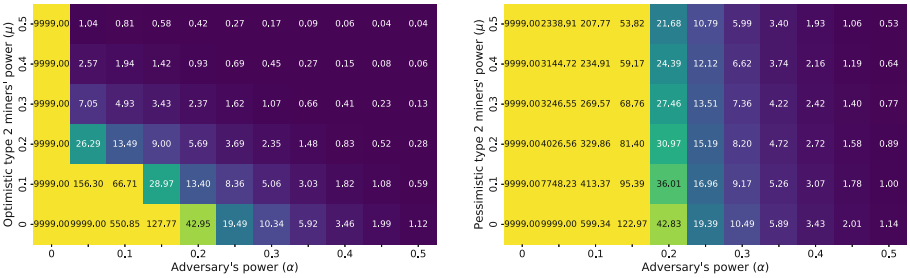
(b) With pessimistic type 2 miners.

Fig. 4. Success rate of the DPC attack depending on the hash power μ of the type 2 miners with the best value of parameter β within 2016 blocks. The "NS" line represents the success rate of the classical double spending attack (based on Nakamoto's evaluation). A darker color indicates a higher success probability.

In presence of type 2 miners (i.e., $\mu > 0$), the DPC attack's success rate is always higher than the one of the traditional double spending attack (i.e., 0(NS) in Fig. 4). The success rate of the double spending attack (with 6 confirmations) with $\alpha = 0.2$ (the power of the biggest mining pool) increases from 1% to 87% (or from 1% to 12%) via the DPC attack depending on μ as shown in Fig. 4a (or Fig. 4b). The impact of optimistic type 2 miners on DPC attack' success rate is more severe than pessimistic type 2 miners, for example, if $\mu = 0.2$ and $\alpha = 0.2$, the DPC attack's success rate is 28% in Fig. 4a while it is 7.7% in Fig. 4b.

Importantly, the DPC attack lowers Bitcoin's safety bound, i.e., the minimum hash power that the adversary needs to double spend or break the chain's consistency. For instance, when $\mu = 0.5, 0.4, 0.3, 0.2$ and type 2 miners are optimistic, a DPC adversary with 30%, 35%, 40%, 45% of the global hash power could completely manipulate the blockchain (i.e., 100% success rate in Fig. 4a), which is more threatening than the existing block withholding attacks [19,21,29].

Inspired by M. Rosenfeld [32], we further evaluate the safe transaction value (i.e., the suggested maximum value of transaction for clients) against double spending attack. Figure 5 plots the minimum value for $\frac{v_t}{v_b}$ that allows the DPC attacker to be more profitable than honest mining. When $\mu = 0.2$ and type 2 miners are optimistic, the adversary with $0.05, 0.1, 0.15, 0.2$ (the possible hash



(a) With optimistic type 2 miners.

(b) With pessimistic type 2 miners.

Fig. 5. Minimum value for $\frac{v_t}{v_b}$ for the DPC attack to be more profitable than honest mining depending on μ. "9999" represents $\frac{v_t}{v_b} \geq 9999$.

power share of mining pools in Bitcoin) of global hash power is incentivized to perform DPC attack as long as the merchants are willing to accept the transaction with $26.29 * v_b, 13.49 * v_b, 9 * v_b, 5.69 * v_b$ BTC respectively (as shown in Fig. 5a). In the same case, when type 2 miners are pessimistic, the safe transaction value would increase and become $4026.56 * v_b, 329.86 * v_b, 81.4 * v_b, 30.97 * v_b$. Bitcoin's future block reward halving will decrease both the threshold to launch profitable DPC attacks and the safe transaction value, which confirms Carlsten et al.'s previous observation [14].

7 Attack Discussion

Attack Variants. We have presented the DPC attack we found to be the most effective when the adversary splits its hash power in two constant parts $\alpha\beta$ and $\alpha(1-\beta)$. We foresee that one could devise variants of the DPC attack, e.g., using techniques that have been applied to selfish mining [19,25,33,34]. In these variants the adversary would mine on different blocks depending on the system's state, or dedicate a different fraction of its hash power to extend each of its two private chains. We leave the study of these variants to future work.

Estimating μ and Selecting β. It is sufficient for the adversary to approximate the value of μ, which is the proportion of the global hash power that belongs to type 2 miners, for a DPC attack to be successful, as our experimental results demonstrate. However, in practice, an adversary would be able to optimize its DPC attack by determining a precise value for μ. The adversary can estimate μ based on some public websites [3], or establish direct connections with the public mining pools to perform a statistical analysis. Moreover, the perishing mining strategy that we present in this paper can be used as a probing technique to measure μ. Indeed, the adversary can directly monitor the impact of perishing mining on the public chain and compute μ based on its growth rate. Once the exact value of μ is known, an adversary can find the best β for the DPC attack by replicating our experiments.

Attack Detection and Prevention. The DPC attack leverages the fact that type 2 miners, which include SPV miners, accept block headers without waiting for and verifying the corresponding transactions. One partial countermeasure against the DPC attack would consist in miners deliberately choosing to stop mining on block headers alone. However, it does not seem reasonable to assume that all miners would avoid this strategy because they can start working on the next block earlier than other miners and therefore increase their profit. Type 2 miners could also avoid mining on the adversary's blocks by accepting to mine only on blocks that were discovered from known mining pools. It is unclear whether this modification would have undesired security implications, e.g., regarding the decentralization of proof-of-work blockchains, or because pool sub-miners run a mining software that is developed internally and independently from the official protocol specification [18]. In addition, this modification would require type 2 miners to trust mining pools, and a malicious pool manager would still be able to execute the DPC attack.

Another idea would be for type 2 miners to stop mining on a block header if the associated transactions are not received before a maximum delay and then mine on the last full block. However, the adversary could also update its strategy to regularly send the unmatched block bodies so that type 2 miners keep mining on its blocks. It is unclear whether this countermeasure would be efficient, and in particular in practical settings. Moreover, the variation of message delays in Bitcoin's peer to peer network would sometimes lead type 2 miners to reject blocks that are generated by honest miners, and might imply possible DoS attacks.

Evidence of Type 2 Mining. In practice, it is difficult to know the exact strategy that miners follow. However, previous works have provided evidence of SPV mining [2–5,27]. Our assumptions in this work are not stronger since our pessimistic type 2 miners are more conservative than SPV miners. In 2020, 9+ mining pools representing 36% of the global power produced empty blocks, which one might consider evidence of SPV mining [1]. We analysed the Bitcoin blockchain and found that Antpool, Binance, F2pool, Huobi, Poolin, ViaBTC published empty blocks from 01/2021 to 02/2022 and collectively represent more than 60% of the global power.

8 Conclusion

In this paper, we proposed perishing mining, a novel adversarial mining strategy that slows down the public chain by leveraging the verifier's dilemma. We then described the dual private chain (DPC) attack where an adversary dedicates a part of its hash power to the perishing mining strategy and launches a parallel double spending attack. We established the Markov decision process of both the perishing mining and the DPC attack. We relied on Monte Carlo simulations to quantify the impact of perishing mining on the public chain growth, and evaluate the double spending success rate of the DPC attack. Our performance evaluation showed that the DPC attack is more powerful than the classical double spending attack as soon as a fraction of the miners mine on blocks without verifying their

transactions. We also evaluated the revenue an adversary could expect from running the DPC attack, and showed that an adversary with sufficient funds or with sufficient hash power would maximize its revenue with the DPC attack.

References

1. Bitcoin miners are mining fewer empty blocks in 2020, and it may not all be due to chance. https://www.theblock.co/post/67928/bitcoin-miners-are-mining-fewer-empty-blocks-in-2020-and-it-may-not-all-because-of-chance. Accessed 2022
2. Empty Blocks. https://medium.com/@ASvanevik/why-all-these-empty-ethereum-blocks-666acbbf002. Accessed 2022
3. f2pool is doing SPV mining. https://bitcointalk.org/index.php?topic=700411.msg11790734#msg11790734. Accessed 2022
4. Half mining power were doing SPV mining. https://bitcoin.org/en/alert/2015-07-04-spv-mining#cause. Accessed 2022
5. SPV mining pools. https://en.bitcoin.it/wiki/Comparison_of_mining_pools. Accessed 2022
6. SPV mining. https://bitcoin.stackexchange.com/questions/38437. Accessed 2022
7. Alharby, M., Lunardi, R.C., Aldweesh, A., van Moorsel, A.: Data-driven model-based analysis of the ethereum verifier's dilemma. In: 2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 209–220. IEEE (2020)
8. Badertscher, C., Lu, Y., Zikas, V.: A rational protocol treatment of 51% attacks. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021. LNCS, vol. 12827, pp. 3–32. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-84252-9_1
9. Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J.A., Felten, E.W.: SOK: research perspectives and challenges for bitcoin and cryptocurrencies. In: SP (2015)
10. Bonneau, J.: Why buy when you can rent? In: Clark, J., Meiklejohn, S., Ryan, P.Y.A., Wallach, D., Brenner, M., Rohloff, K. (eds.) FC 2016. LNCS, vol. 9604, pp. 19–26. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53357-4_2
11. Cao, T., Decouchant, J., Yu, J.: Leveraging the verifier's dilemma to double spend in bitcoin. arXiv preprint arXiv:2210.14072 (2022)
12. Cao, T., Decouchant, J., Yu, J., Esteves-Verissimo, P.: Characterizing the impact of network delay on bitcoin mining. In: 2021 40th International Symposium on Reliable Distributed Systems (SRDS), pp. 109–119. IEEE (2021)
13. Cao, T., Yu, J., Decouchant, J., Luo, X., Verissimo, P.: Exploring the monero peer-to-peer network. In: Bonneau, J., Heninger, N. (eds.) FC 2020. LNCS, vol. 12059, pp. 578–594. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-51280-4_31
14. Carlsten, M., Kalodner, H., Weinberg, S.M., Narayanan, A.: On the instability of bitcoin without the block reward. In: CCS (2016)
15. Croman, K., et al.: On scaling decentralized blockchains. In: Clark, J., Meiklejohn, S., Ryan, P.Y.A., Wallach, D., Brenner, M., Rohloff, K. (eds.) FC 2016. LNCS, vol. 9604, pp. 106–125. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53357-4_8
16. Decker, C., Wattenhofer, R.: Information propagation in the bitcoin network. In: IEEE P2P (2013)
17. Dembo, A., et al.: Everything is a race and Nakamoto always wins. In: CCS (2020)

18. Eyal, I.: The miner's dilemma. In: IEEE Symposium on Security and Privacy (2015)
19. Eyal, I., Sirer, E.G.: Majority is not enough: bitcoin mining is vulnerable. In: Christin, N., Safavi-Naini, R. (eds.) FC 2014. LNCS, vol. 8437, pp. 436–454. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-45472-5_28
20. Gaži, P., Kiayias, A., Russell, A.: Tight consistency bounds for bitcoin. In: CCS (2020)
21. Gervais, A., Karame, G.O., Wüst, K., Glykantzis, V., Ritzdorf, H., Capkun, S.: On the security and performance of proof of work blockchains. In: CCS (2016)
22. Göbel, J., Keeler, H.P., Krzesinski, A.E., Taylor, P.G.: Bitcoin blockchain dynamics: the selfish-mine strategy in the presence of propagation delay. Perform. Eval. **104**, 23–41 (2016)
23. Han, R., Sui, Z., Yu, J., Liu, J.K., Chen, S.: Fact and fiction: challenging the honest majority assumption of permissionless blockchains. In: ASIA CCS (2021)
24. Heilman, E., Kendler, A., Zohar, A., Goldberg, S.: Eclipse attacks on bitcoin's peer-to-peer network. In: USENIX Security (2015)
25. Karame, G.O., Androulaki, E., Capkun, S.: Double-spending fast payments in bitcoin. In: CCS (2012)
26. Luu, L., Teutsch, J., Kulkarni, R., Saxena, P.: Demystifying incentives in the consensus computer. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, pp. 706–719 (2015)
27. Mirkin, M., Ji, Y., Pang, J., Klages-Mundt, A., Eyal, I., Juels, A.: BDoS: blockchain denial-of-service. In: CCS (2020)
28. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008)
29. Nayak, K., Kumar, S., Miller, A., Shi, E.: Stubborn mining: generalizing selfish mining and combining with an eclipse attack. In: Euro S& P (2016)
30. Negy, K.A., Rizun, P.R., Sirer, E.: Selfish mining re-examined. In: Bonneau, J., Heninger, N. (eds.) FC 2020. LNCS, vol. 12059, pp. 61–78. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-51280-4_5
31. Pass, R., Seeman, L., Shelat, A.: Analysis of the blockchain protocol in asynchronous networks. In: Coron, J.-S., Nielsen, J.B. (eds.) EUROCRYPT 2017. LNCS, vol. 10211, pp. 643–673. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-56614-6_22
32. Rosenfeld, M.: Analysis of bitcoin pooled mining reward systems. CoRR (2011)
33. Rosenfeld, M.: Analysis of hashrate-based double spending. arXiv preprint arXiv:1402.2009 (2014)
34. Sapirshtein, A., Sompolinsky, Y., Zohar, A.: Optimal selfish mining strategies in bitcoin. In: Grossklags, J., Preneel, B. (eds.) FC 2016. LNCS, vol. 9603, pp. 515–532. Springer, Heidelberg (2017). https://doi.org/10.1007/978-3-662-54970-4_30
35. Sompolinsky, Y., Zohar, A.: Bitcoin's security model revisited. arXiv preprint arXiv:1605.09193 (2016)
36. Teutsch, J., Reitwießner, C.: A scalable verification solution for blockchains. arXiv preprint arXiv:1908.04756 (2019)
37. Yu, J., Kozhaya, D., Decouchant, J., Verissimo, P.: Repucoin: your reputation is your power. IEEE Trans. Comput. (2019)