BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2023
http://excel.fit.vutbr.cz

Fee-Redistribution Contracts

Rastislav Budinský

Abstract

We originally reviewed the undercutting attacks in the transaction-fee-based regime of proof-of-work (PoW) blockchains with the longest chain fork-choice rule. Next, we focused on the problem of fluctuations in mining revenue and the mining gap – i.e., a situation, in which the immediate reward from transaction fees does not cover miners' expenditures.

To mitigate these issues, we propose a solution that splits transaction fees (however our solution is not applicable exclusively to transaction fees) from a mined block into two parts – (1) an instant reward for the miner of a block and (2) a deposit sent to one or more fee-redistribution contracts ($\mathcal{FRC}$s) that are part of the consensus protocol. At the same time, these redistribution contracts reward the miner of a block with a certain fraction of the accumulated funds of the incoming fees over a predefined time. This setting enables us to achieve several interesting properties that are beneficial for the incentive stability and security of the protocol.

With our solution, the fraction of Default-Compliant miners who strictly do not execute undercutting attack is lowered from the state-of-the-art result of 66% to 30%.

xbudin05@fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

[Motivation] Carlsten et al. [1] pointed out the effects of the high variance of the miners' revenue per block caused by exponentially distributed block arrival time in transaction-fee-based protocols. The authors showed that *undercutting* (i.e., forking) a wealthy block is a profitable strategy for a malicious miner.

Nevertheless, Daian et al. [2] showed that this attack is viable even in blockchains containing traditional block rewards due to front-running competition of arbitrage bots who are willing to extremely increase transaction fees to earn Maximum Extractable Value profits.

Our work originally focused on the following three problems.

[Problem definition]

Mining gap Makes it unprofitable for miner to mine right after a new block was mined as it included most of the transactions from his mempool.

Fluctuation of miner's revenue Is a situation, in which over the span of certain time frame the rewards from transactions vary leading to discrepancies in rewards from the mined blocked.

Undercutting attack Is performed by re-mining top block with wealthy collected fees in order to keep fraction of those fees and leaving the rest to incentivize the next miner to mine on this new block.

[Existing solutions] Gong et al. [3] model the profitability of undercutting with the block size limit presented, which bounds the claimable fees in a mining round. The authors presented a countermeasure that selectively assembles transactions into the new block, while claiming fewer fees to avoid undercutting.

[Our solution] You can have a quick look at Figure 1. displaying the basic scheme we propose. We introduce a contract, which collects percentage of collected fees and redistributes them at the same time back, while acting as a sliding window averaging function for the reward.

[Contributions] Our solution is simple yet effective against the introduced problems in eliminating them or significantly reducing their impact. It is already being discussed as improvement proposal on Ergo blockchain for redistributing Storage Rent Fees utilizing Fee-Redistribution Contracts, please see link.

2. Fee-Redistribution Contracts

Our proposed solution collects a percentage from all transaction fees, which are usually paid in a native cryptocurrency coin, collected in the mined blocks into one or multiple fee-redistribution contracts (i.e., $\mathcal{FRC}$s). Miners of the blocks, who must contribute to these contracts, are at the same time rewarded from them, while the received reward approximates a moving average of the incoming transaction fees across the fixed sliding window of the blocks.

The fraction of transaction fees (i.e., $\mathbb{C}$) from the mined block is sent to $\mathcal{FRC}$s and the remaining fraction of transaction fees (i.e., $\mathbb{M}$) is directly assigned to the miner, such that $\mathbb{C}+\mathbb{M}=1$.

The role of $\mathbb{M}$ is to keep the incentive for the miners to prioritize transactions with higher feerate in tack in order to keep the free-market biding feature unchanged. Leaving this feature for users creating transactions as option to have a chance to bid higher feerate in order to have their transaction included in sooner block.

While the role of $\mathbb{C}$ is to mitigate the problems, this paper is focused on, such as fluctuating miner's revenue, undercutting attacks and the mining gap by averaging the collected fees. And rewarding the miner with averaged reward over different time period, based on individual contracts. Roughly said a miner can expect $\mathbb{M}$ of his reward directly from the collected fees and $\mathbb{C}$ as averaged reward of fees collected in individual contracts.

2.1 Defining Fee-Redistribution Contracts

We define a Fee-Redistribution Contract as following tuple $\mathcal{FRC}=(\nu,\lambda,\rho)$ where

- ν is the accumulated amount of usually native blockchain coins in the contract &
- λ denotes the size of $\mathcal{FRC}$' sliding window in terms of the number of preceding blocks that contributed to ν &
- And ρ is the parameter defining the ratio for redistribution of incoming collected transaction fees to the particular $\mathcal{FRC}$ among multiple contracts, while the sum of ρ across all $\mathcal{FRC}$s must be equal to 1 (i.e. see Equation 1).

$$\sum_{x \in \mathcal{FRC}s} x.\rho \quad = \quad 1. \tag{1}$$

3. Overview

We depict the overview of our approach in Figure 1., and it consists of the following steps:

1. Using $\mathcal{FRC}$, the miner calculates the reward for the next block B he receives from the (i.e., $nextClaim(\mathcal{FRC})$ – see Equation 3) that will be payed by $\mathcal{FRC}$ to the miner of that block.
2. The miner mines the block B using the selected set of transactions with the highest feerate to maximize the profit from his mempool.
3. The miner of the mined block B directly receives a certain fraction of all the collected transaction fees (i.e., $B.fees * \mathbb{M}$) and the remaining part (i.e., $B.fees * \mathbb{C}$) the miner sends to $\mathcal{FRC}$.
4. The miner obtains $nextClaim$ from $\mathcal{FRC}$.

Our approach is embedded into the consensus protocol, and therefore consensus nodes are obliged to respect it in order to ensure that their blocks are valid, accepted by the network resulting in receiving rewards. It can be implemented with standard smart contracts of the blockchain platform or within the native code of the consensus protocol. In the environment with constant transaction fees a miner would receive the same amount with or without our solution.

3.1 Reward from Contracts – $nextClaim$

$$\partial Claim^{\mathcal{FRC}_{[H]}}_{[H+1]} = \frac{\mathcal{FRC}_{[H]}.\nu}{\mathcal{FRC}_{[H]}.\lambda}, \tag{2}$$

$$nextClaim_{[H+1]} = \sum_{\mathcal{X}_{[H]} \in \mathcal{FRC}s_{[H]}} \partial Claim^{\mathcal{X}_{[H]}}_{[H+1]}. \tag{3}$$

4. Results & Conclusion

We summarize the results of our work (older version can be found here) briefly.

- Multiple Contract setup provides higher flexibility in averaging the rewards in most scenarios.
- Longer Contracts hold reward more steadily while short-term contract fluctuate more. However shorter contracts better reflect miners' expectations.
- Mining gap problem is greatly reduced with proper $\mathbb{C}$&$\mathbb{M}$ parameters.
- Fluctuation of miner's revenue is averaged by contracts, therefore the problem is mitigated. Summarized from experiments under Figure 2..
- Undercutting attack is also greatly reduced, with $\mathbb{C}=0.7$ and $\mathbb{M}=0.3$ respectively the number of honest miners to prevent the attack falls from 66% to 30%. This is displayed in Figure 4..

References

[1] Miles Carlsten, Harry Kalodner, S Matthew Weinberg, and Arvind Narayanan. On the instability of bitcoin without the block reward. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 154–167, 2016.

[2] Philip Daian, Steven Goldfeder, Tyler Kell, Yunqi Li, Xueyuan Zhao, Iddo Bentov, Lorenz Breidenbach, and Ari Juels. Flash boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 910–927. IEEE, 2020.

[3] Tiantian Gong, Mohsen Minaei, Wenhai Sun, and Aniket Kate. Towards overcoming the undercutting problem. In *International Conference on Financial Cryptography and Data Security*, pages 444–463. Springer, 2022.