RPPLNS: Pay-per-last-N-shares with a Randomised Twist

Extended Abstract

Philip Lazos
Sapienza University of Rome
lazos@diag.uniroma1.it

Francisco J. Marmolejo-Cossío
University of Oxford and IOHK
francisco.marmolejo@cs.ox.ac.uk

Xinyu Zhou
University of Maryland
xyzhou@terpmail.umd.edu

Jonathan Katz
University of Maryland
jkatz2@gmail.com

ABSTRACT

"Pay-per-last-N-shares" (PPLNS) is one of the most common payout strategies used by mining pools in Proof-of-Work (PoW) cryptocurrencies such as Bitcoin. As with any payment scheme, it is imperative to study issues of incentive compatibility of miners within the pool. For PPLNS this question has only been partially answered; we know that reasonably-sized miners within a PPLNS pool prefer following the pool protocol over employing *specific* deviations. In this paper, we present a novel modification to PPLNS where we randomise the protocol in a natural way. We call our protocol "Randomised pay-per-last-N-shares" (RPPLNS), and note that the randomised structure of the protocol greatly simplifies the study of its incentive compatibility. We show that RPPLNS maintains the strengths of PPLNS (i.e., fairness, variance reduction, and resistance to pool hopping), while also being robust against a richer class of strategic mining than what has been shown for PPLNS.

KEYWORDS

blockchain; strategic mining; mining pools

ACM Reference Format:
Philip Lazos, Francisco J. Marmolejo-Cossío, Xinyu Zhou, and Jonathan Katz. 2021. RPPLNS: Pay-per-last-N-shares with a Randomised Twist: Extended Abstract. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 3 pages.

1 INTRODUCTION

In Bitcoin, miners expend computational resources to maintain a ledger of transactions for all users in the system. This is done by successfully mining "blocks" of transactions, for which miners earn a reward. The nature of mining blocks is inherently variable. Hence, miners often form groups and pool their resources so that rather than rarely earning large rewards they earn smaller rewards at a more consistent rate. In order to share rewards, said pools must reliably identify the computational contribution of each miner. This is done by accepting "near-misses" (called shares) to Bitcoin's desired hash rate, created at a rate directly proportional to the computational effort spent on extending the blockchain. A pool operator collects shares reported by every miner in the pool, and uses them to distribute payments once an actual block is found. Participating in a pool should have (at least) the following guarantees:

(1) Fairness: miners earn the same block reward in expectation as mining alone.
(2) Variance reduction: miners have lower variance in block reward than when mining alone.
(3) Robustness against pool hopping: at no point of time does the miner benefit from switching between similar pools.
(4) Incentive Compatibility: to maximize their reward, each participating miner should always expend maximum effort and report shares/blocks immediately as they are generated.

One of the most popular pool mining protocols which (partially) satisfies these properties is "Pay-per-last-N-shares" (PPLNS). Miners report shares to a pool operator which maintains a queue of the N most recent shares reported to it, and if a block is found and reported by the pool, the owners of these N shares are paid proportionally ($1/N$ times the value of a block for each such share). The structure of PPLNS is such that it satisfies properties 1-3 above [13]. With respect to property 4, [15, 16] demonstrate that miners in PPLNS pools are incentivised to act honestly if they are only permitted specific deviations, hence PPLNS only partially fulfils it.

Our Contributions. We modify PPLNS so that a pool operator maintains a "bag" of N shares and when a new share is found, it replaces a random share from the bag. We call our protocol "Randomised pay-per-last-N-shares" (RPPLNS), and note that the randomised structure greatly simplifies the study of its incentive compatibility. We show[1] that RPPLNS maintains the strengths of PPLNS, and we use experimental evidence to show that RPPLNS is robust to strategic pool miners who may *arbitrarily* hoard share and blocks to publish at a later time.

Related Work. Strategic mining has been studied since the inception of Bitcoin [10]. In [4], the authors demonstrate that honest mining is not robust to strategic mining in terms of block reward, even when a miner has less than a majority computational stake in the Bitcoin ecosystem. This work is refined in [14], [11] and [6] by generalising selfish mining, pairing selfish mining with network-level attacks, and proving limited incentive compatibility of honest mining at low hash rates. Further incentives at the individual miner level have also been studied in [2, 7, 9]. At the pool level, the authors of [3] and [8] study infiltration attacks pools can wage against

Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, Online.

[1] The full version of the paper can be found in [5].

each other which lead to an iterated prisoner's dilemma between operators and [1] study reward sharing for stake pools.

An extensive survey of pool protocols can be found in [13]. In [15], the authors study incentive compatibility in pool protocols that decide how to make payments on the basis of the quantity of shares each miner reported, irrespective of the order in which they are received. They also study PPLNS and show that honest pool mining is robust to specific share/block hoarding strategies whereby miners enter a hoarding phase for shares, and simply publish all secret shares upon finding a new block. The authors of [16] study a similar set of strategic deviations where for a fixed $x \in \mathbb{N}$ a miner hoards at most x shares. Subsequent shares are published immediately, and whenever a block is found, those x shares are published immediately before publishing the block. Their analysis makes the assumption that each strategic miner reaches their threshold x, and show when being honest outperforms being strategic in this setting. The authors of [12] find reporting strategies that can be beneficial to strategic miners at high enough hash rates.

2 PROPERTIES OF RPPLNS

To analyse RPPLNS, it suffices to consider a single strategic pool miner, m_1 with hash power α, a single honest pool miner m_2 with hash power β, and a single honest non-pool miner m_0 with hash power γ. Indeed, m_2 and m_0 could be composed of multiple miners, but if they are honest, we can model their behaviour as that of a single miner of their aggregate hash power. Pool miners operate within an RPPLNS pool with a bag of size N, and such that the block difficulty threshold is D times higher than that of shares. To model revenues, we consider a turn-based process. Every turn, either m_1, m_2 or m_0 find a share with probability α, β and γ respectively, and each share has a further $\frac{1}{D}$ probability of being a full block. m_0 finds shares in the sense that it computes a block with a hash that is a near-miss to the target hash (by a factor of D), but does not actually report this near miss to the pool since it is not a part of the pool. m_0 does publish blocks immediately to all agents of the Bitcoin ecosystem, including the pool operator. m_1 may hoard shares/blocks found, but since m_2 is an honest pool miner, whenever they find a share/block they communicate this immediately to the RPPLNS pool. The full version can be found in [5].

Fairness, Variance Reduction and Pool Hopping. In the full version of the paper, we show that if m_1 is honest, then their expected block reward per turn is precisely α/D. Since each share has a $\frac{1}{D}$ probability of being a block, this coincides with the expected α block reward m_1 would get (per block mined by the system) by mining solo. In addition, we demonstrate that RPPLNS enjoys similar variance reduction in block reward to that of PPLNS.

Theorem 2.1. *Suppose that m_1 is honest with hash power α. Their expected per-turn block reward in a RPPLNS pool is $\frac{\alpha}{D}$ and the variance of the reward is $\frac{1}{D^2}(\alpha - \alpha^2) + \frac{\alpha}{ND}$.*

In deterministic PPLNS, block reward variance can be computed in an identical fashion, and it is $\frac{\alpha}{2D^2} + \frac{\alpha}{ND} - \frac{\alpha^2}{D^2} - \frac{\alpha}{2ND^2}$. Typically, pools have $N = 2D$, in which case the PPLNS variance becomes $\frac{1}{D^2}(\alpha - \alpha^2) - \frac{\alpha}{4D^3}$. For this difficulty setting, RPPLNS block reward variance becomes $\frac{1}{D^2}(\alpha - \alpha^2) + \frac{\alpha}{2D^2}$. Though this is more than with standard PPLNS, this still vanishes at the same asymptotic rate of $O(1/N^2)$ when $N = \Theta(D)$. Finally, we show that for a given time horizon, a pool miner's reward only depends on total effort invested in the pool, and not on *when* the effort is invested, which leads to:

Theorem 2.2 (Informal). *RPPLNS is resistant to pool hopping.*

When is Honest Mining a Dominant Strategy. We wish to find conditions such that m_1 is honest. To this end, in the full paper, we provide a recurrence that bounds the block reward that m_1 can obtain from being optimally strategic. We compute the best possible reward of m_1, assuming $N = 1000, D = 500$ and a finite horizon of $k = 150$ steps. We compare against expected rewards in the same number of steps if m_1 follows the protocol honestly. The following graphs show the best action for m_1 given an *initial* fraction of F shares in the bag. We do not include graphs for $F \in (0.33, 0.7)$ as m_1 is only honest in these cases. We witness some strategic behaviour, though it is important to note that this occurs at hash rates of m_1 where having the given initial fraction of F shares is improbable. In the full version of the paper, we justify the strategic deviations we see, and prove that PPLNS is susceptible to the same deviations at similarly improbable extremal queue states.

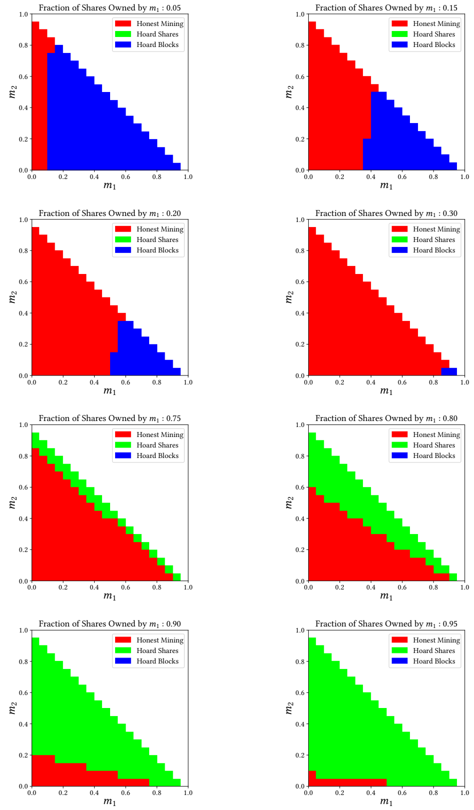

Figure 1: $F \leq 0.30$ **and** $0.75 \leq F \leq 0.95$

ACKNOWLEDGMENTS

Philip Lazos is supported by the ERC Advanced Grant 788893 AMDROMA "Algorithmic and Mechanism Design Research in Online Markets" and MIUR PRIN project ALGADIMAR "Algorithms, Games, and Digital Markets"

REFERENCES

[1] Lars Brünjes, Aggelos Kiayias, Elias Koutsoupias, and Aikaterini-Panagiota Stouka. 2020. Reward Sharing Schemes for Stake Pools. In *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 256–275.

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

[3] Ittay Eyal. 2015. The miner's dilemma. In *2015 IEEE Symposium on Security and Privacy*. IEEE.

[4] Ittay Eyal and Emin Gün Sirer. 2014. Majority is not enough: Bitcoin mining is vulnerable. In *International conference on financial cryptography and data security*. Springer, 436–454.

[5] Jonathan Katz, Philip Lazos, Francisco J. Marmolejo-Cossío, and Xinyu Zhou. 2021. RPPLNS: Pay-per-last-N-shares with a Randomised Twist. arXiv:2102.07681 [cs.GT]

[6] Aggelos Kiayias, Elias Koutsoupias, Maria Kyropoulou, and Yiannis Tselekounis. 2016. Blockchain mining games. In *Proceedings of the 2016 ACM Conference on Economics and Computation*. ACM, 365–382.

[7] Elias Koutsoupias, Philip Lazos, Foluso Ogunlana, and Paolo Serafino. 2019. Blockchain Mining Games with Pay Forward. In *The World Wide Web Conference*. 917–927.

[8] Yujin Kwon, Dohyun Kim, Yunmok Son, Eugene Vasserman, and Yongdae Kim. 2017. Be selfish and avoid dilemmas: Fork after withholding (faw) attacks on bitcoin. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 195–209.

[9] Francisco J Marmolejo-Cossío, Eric Brigham, Benjamin Sela, and Jonathan Katz. 2019. Competing (Semi)-Selfish Miners in Bitcoin. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*. 89–109.

[10] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system.

[11] Kartik Nayak, Srijan Kumar, Andrew Miller, and Elaine Shi. 2016. Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. In *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*. IEEE, 305–320.

[12] Rui Qin, Yong Yuan, and Fei-Yue Wang. 2019. A novel hybrid share reporting strategy for blockchain miners in PPLNS pools. *Decision Support Systems* 118 (2019), 91–101.

[13] Meni Rosenfeld. 2011. Analysis of bitcoin pooled mining reward systems. *arXiv preprint arXiv:1112.4980* (2011).

[14] Ayelet Sapirshtein, Yonatan Sompolinsky, and Aviv Zohar. 2016. Optimal selfish mining strategies in bitcoin. In *International Conference on Financial Cryptography and Data Security*. Springer, 515–532.

[15] Okke Schrijvers, Joseph Bonneau, Dan Boneh, and Tim Roughgarden. 2016. Incentive compatibility of bitcoin mining pool reward functions. In *International Conference on Financial Cryptography and Data Security*. Springer, 477–498.

[16] Yevhen Zolotavkin, Julian García, and Carsten Rudolph. 2017. Incentive compatibility of pay per last N shares in bitcoin mining pools. In *International Conference on Decision and Game Theory for Security*. Springer, 21–39.