Trustless and Bias-resistant Game-theoretic Distributed Randomness

Zhuo Cai, Amir Kafshdar Goharshady

Trustless and Bias-resistant Game-theoretic Distributed Randomness

Zhuo Cai
Department of Computer Science and Engineering
Hong Kong University of Science and Technology
Hong Kong SAR, China
zcaiam@connect.ust.hk

Amir Kafshdar Goharshady
Department of Computer Science and Engineering
Hong Kong University of Science and Technology
Hong Kong SAR, China
goharshady@ust.hk

***Abstract*—Proof-of-Stake blockchain protocols rely on a distributed random beacon to select the next miner that is allowed to add a block to the chain. Each party's likelihood to be selected is in proportion to their stake in the cryptocurrency. Current random beacons used in PoS protocols have two fundamental limitations: either (i) they rely on pseudo-randomness, e.g. assuming that the output of a hash function is uniform, which is an unproven assumption, or (ii) they generate their randomness using a distributed protocol in which several participants are required to submit random numbers which are then used in the generation of a final random result. However, in this case, there is no guarantee that the numbers provided by the parties are truly random and there is no incentive for the parties to honestly generate uniform randomness.**

In this work, we provide a protocol that generates trustless and unbiased randomness for PoS and overcomes the above limitations. We provide a game-theoretic guarantee showing that it is in everyone's best interest to submit truly uniform random numbers. Hence, our approach is the first to provably incentivize honest and reliable behavior instead of simply assuming it.

I. Introduction

Proof-of-work is quite inefficient and uses huge amounts of energy. Thus, many alternatives are designed in the literature to ameliorate this problem [1]–[5]. The most prominent alternative is proof-of-stake (PoS). In PoS blockchain protocols, miners are chosen randomly and each miner's chance of being allowed to add the next block should be proportional to their stake in the currency. The security claims of proof-of-stake protocols rely on the assumption that a majority, or a high percentage, of the stake is owned by honest participants. Despite their differences, all proof-of-stake protocols require a random beacon to randomly select the next miners. As an example, Ouroboros [3] uses a publicly verifiable secret sharing scheme (PVSS [6]) to generate a random seed for each epoch. However, in this scheme the participants have no incentive to submit a uniform random value. Ouroboros Praos [2] and Algorand [7] update the random seed by applying verifiable random functions (VRF [8], [9]) to blockchain data in previous rounds. A major drawback of this randomness beacon is that the generated numbers are not guaranteed to be uniform.

Besides the PoS protocols, smart contracts also use decentralized randomness in DeFi. For example, RANDAO [10] is a family of smart contracts that produce random numbers. Anyone can participate and submit a random value to contribute to the output. It uses a commitment scheme where participants submit a cryptographically hashed commitment in the first phase and then reveal the value in the second phase. Participants are incentivized to be *honest*, which means following the protocol and sending valid messages in time, because they will lose their deposits otherwise. However, a malicious party can bias the output by choosing to not reveal their value. Moreover, RANDAO also fails to incentivize participants to be *reliable*, meaning submitting uniformly random values. Incentivizing reliability is essential to guarantee uniform randomness of the output.

In this work, we design a novel game-theoretic approach for randomness generation, which builds upon game-theoretic ideas in [11]–[14]. We call this an RIG (Random Integer Generation) game. RIG efficiently produces a uniform random integer from an arbitrarily large interval. Moreover, the only equilibrium in an RIG is for all participants to choose their values uniformly at random. Finally, RIG can be plugged into common proof-of-stake blockchains with minor changes and negligible overhead.

Our protocols are the first to incentivize participants to be reliable and submit truly uniform random numbers. In comparison, previous distributed randomness protocols [15]–[18] using commitment schemes and PVSS assume that there is at least one reliable participant without incentivizing reliability. In other words, they only reward honesty but assume both honesty and reliability. The reliability assumption is unfounded. Several other randomness protocols, including Algorand and Ouroboros Praos, do not depend on random inputs from participants at all, but instead use real-time data on blockchains and cryptographic hash functions to generate pseudo-random numbers. This pseudo-randomness is not guaranteed to be uniform. Hence, there is no guarantee that miners get elected with probabilities proportional to their stake.

II. Random Integer Generation Game (RIG)

$(n, m)-$***RIG.*** Suppose that we have n players and n is even. A *Random Integer Generation game* (RIG) with n players and $m \geq 3$ strategies is a game G in which:

- For every player $i \in \{1, \ldots, n\}$, we have m pure strategies $S_i = \{0, 1, \ldots, m-1\}$;
- We pair the players such that every even player is paired with the previous odd player and every odd player is paired with the next even player. In other words, $pair(2 \cdot k) = 2 \cdot k - 1$ and $pair(2 \cdot k - 1) = 2 \cdot k$.

- At an outcome $s = (s_1, s_2, \ldots, s_n)$ of the game, the payoff of player i is defined as $u_i(s) \coloneqq f(s_i, s_j)$ where $j = pair(i)$, and

$$f(s_i, s_j) \coloneqq \begin{cases} 1 & \text{if } s_i - s_j \equiv 1 \pmod m \\ -1 & \text{if } s_i - s_j \equiv -1 \pmod m \\ 0 & \text{otherwise} \end{cases}$$

Essentially, we assume that any adjacent pair of even player and odd player play a zero-sum symmetric one-shot game with each other. Their payoffs are independent of other $n-2$ players. For each pair $(2 \cdot k - 1, 2 \cdot k)$ of players, this is a zero-sum matrix game with the following payoff matrix:

$$A = \begin{pmatrix} 0 & -1 & 0 & \cdots & 0 \\ 1 & 0 & -1 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix}$$

Theorem 1: (Alliance-Resistant Nash Equilibrium of an RIG.) Let G be an RIG game with n players and m strategies, where n is an even number and $m \geq 3$. Let $\bar{\sigma}$ be a mixed strategy profile defined by $\bar{\sigma}_i = (1/m, 1/m, \ldots, 1/m)$ for all i, i.e. the mixed strategy profile in which each player i chooses a strategy in S_i uniformly at random. Then, $\bar{\sigma}$ is the *only* Nash equilibrium of G. Further, it is also alliance-resistant, i.e. no subset of players can collaborate to increase their overall utility.

The theorem above shows that it is in every player's best interest to play uniformly at random, i.e. choose each pure strategy in S_i with probability exactly $1/m$. Moreover, this equilibrium is self-enforcing even in the presence of alliances. Hence, we can plug this game into a distributed random number generation protocol and give participants rewards that are based on their payoffs in this game. This ensures that every participant is incentivized to provide a uniformly random s_i. As mentioned before, even if only one participant is reliable and submits a uniformly random s_i, then the entire result $s = \sum_{i=1}^{n} s_i \pmod m$ of the random number generation protocol is guaranteed to be unbiased. Hence, instead of assuming that a reliable party exists, we incentivize every party to be reliable.

III. Designing a Random Beacon Based on RIG

We assume a synchronous communication model, where every message is delivered after an at most constant delay. We also consider rational players. We discuss two schemes to execute the RIG game in a PoS protocol: (1) using commitment schemes and verifiable delay functions (VDF [19]), and (2) using PVSS [6].

The first scheme includes a commit phase and a reveal phase as usual. There is an additional VDF phase, to ensure that malicious participants cannot bias the output by choosing to not reveal the values. However, VDFs are not proved to preserve uniformity of randomness. Even if the sum $s = \sum_{i=1}^{n} s_i \pmod m$ is uniformly random, the result VDF(s) might not be uniformly random. Therefore, we propose to use $s' = s_1 + \text{VDF}(s_2)$ as the final random output, where s_1 and s_2 are lower and higher bits of s, respectively. s' is not only unpredictable at the reveal phase but also uniformly random.

The second scheme consists of a share distribution phase and a reconstruction phase. PVSS ensures that malicious participants cannot bias by not revealing without VDFs. Instead, PVSS can force the opening of each value by reconstructing from its secret shares as long as more than half of the participants are honest to decrypt secret shares.

In contrast to RANDAO and many other random number generators, our RIG game is sensitive to the order of participants. The result of the RIG game is not only the output value, but also the payoffs. We can use the random seed in the previous epoch to randomly sort the participants.

Honest participants should have positive expected payoffs that at least cover the communication cost on chain, to be incentivized to participate the RIG game. Dishonest participants are excluded from the game output and lose their deposits.

IV. RIG in Proof of Stake Protocols

We now show how our RIG random beacon can supplant standard PoS protocols. In general, the RIG random beacon, be it implemented by the commitment scheme approach or the PVSS approach, is applicable to any PoS protocol that requires an evolving random seed to select miners.

In Ouroboros Praos. Ouroboros Praos is the underlying protocol of Cardano cryptocurrency. We can substitute the random beacon of Ouroboros Praos with our RIG. In Cardano, 1 epoch lasts for 5 days, and the transaction confirmation time is 20 minutes. When using the commitment scheme, we require more time ($\approx 3\times$ transaction confirmation time) within an epoch for the extra reveal phase and VDF computation time to reach a consensus on the result of RIG, which is negligible.

In Algorand. We can use RIG based on PVSS as the random beacon of Algorand. Algorand reaches consensus within 1 round, and updates the random seed once every 1000 rounds, which is sufficient for a PVSS execution. Moreover, assuming that 100 participants join the RIG game, using RIG decreases the transaction-per-second by less than 1%.

V. Conclusion

In this work, we presented a game-theoretic beacon for distributed random number generation. We showed that our approach is bias-resistant, unpredictable, available, and verifiable. Moreover, it incentivizes every participant to be reliable, i.e. provide a truly uniform random input. Even if only one of the participants is rational and therefore reliable, the output number is guaranteed to be sampled from the uniform distribution and to be unbiased. Additionally, our approach does not use pseudo-randomness at any point and instead only relies on well-incentivized game-theoretic randomness. Finally, even though the approach is general and not limited to blockchain use cases, we showed that one can easily augment common proof-of-stake protocols to include our randomness beacon for the task of selecting their miners. This ensures that proof-of-stake protocols choose the miners fairly, i.e. exactly in proportion to their stake.

Acknowledgment

The research was partially supported by the Hong Kong Research Grants Council ECS Project Number 26208122, the HKUST-Kaisa Joint Research Institute Project Grant HKJRI3A-055 and the HKUST Startup Grant R9272.

REFERENCES

[1] S. Park, A. Kwon, G. Fuchsbauer, P. Gazi, J. Alwen, and K. Pietrzak, "SpaceMint: A cryptocurrency based on proofs of space," in *FC*, 2018, pp. 480–499.

[2] B. David, P. Gaži, A. Kiayias, and A. Russell, "Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain," in *CRYPTO*, 2018, pp. 66–98.

[3] A. Kiayias, A. Russell, B. David, and R. Oliynykov, "Ouroboros: A provably secure proof-of-stake blockchain protocol," in *CRYPTO*, 2017, pp. 357–388.

[4] M. A. Meybodi, A. K. Goharshady, M. R. Hooshmandasl, and A. Shakiba, "Optimal mining: Maximizing Bitcoin miners' revenues from transaction fees," in *IEEE Blockchain*, 2022, pp. 266–273.

[5] K. Chatterjee, A. K. Goharshady, and A. Pourdamghani, "Hybrid mining: Exploiting blockchain's computational power for distributed problem solving," in *SAC*, 2019, pp. 374–381.

[6] B. Schoenmakers, "A simple publicly verifiable secret sharing scheme and its application to electronic voting," in *CRYPTO*, 1999, pp. 148–164.

[7] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich, "Algorand: Scaling Byzantine agreements for cryptocurrencies," in *SOSP*, 2017, pp. 51–68.

[8] S. Micali, M. O. Rabin, and S. P. Vadhan, "Verifiable random functions," in *FOCS*, 1999, pp. 120–130.

[9] Y. Dodis and A. Yampolskiy, "A verifiable random function with short proofs and keys," in *PKC*, 2005, pp. 416–431.

[10] "RANDAO: A DAO working as RNG of Ethereum," 2019. [Online]. Available: https://github.com/randao/randao

[11] K. Chatterjee, A. K. Goharshady, R. Ibsen-Jensen, and Y. Velner, "Ergodic mean-payoff games for the analysis of attacks in crypto-currencies," in *CONCUR*, 2018.

[12] K. Chatterjee, A. K. Goharshady, and Y. Velner, "Quantitative analysis of smart contracts," in *ESOP*, 2018, pp. 739–767.

[13] K. Chatterjee, A. K. Goharshady, and A. Pourdamghani, "Probabilistic smart contracts: Secure randomness on the blockchain," in *ICBC*, 2019, pp. 403–412.

[14] A. K. Goharshady, "Irrationality, extortion, or trusted third-parties: Why it is impossible to buy and sell physical goods securely on the blockchain," in *IEEE Blockchain*, 2021, pp. 73–81.

[15] E. Syta, P. Jovanovic, E. Kokoris-Kogias, N. Gailly, L. Gasser, I. Khoffi, M. J. Fischer, and B. Ford, "Scalable bias-resistant distributed randomness," in *SP*, 2017, pp. 444–460.

[16] P. Schindler, A. Judmayer, N. Stifter, and E. R. Weippl, "Hydrand: Efficient continuous distributed randomness," in *SP*, 2020, pp. 73–89.

[17] G. Wang and M. Nixon, "Randchain: Practical scalable decentralized randomness attested by blockchain," in *IEEE Blockchain*, 2020, pp. 442–449.

[18] M. Krasnoselskii, G. Melnikov, and Y. Yanovich, "No-dealer: Byzantine fault-tolerant random number generator," in *INFOCOM*, 2020, pp. 568–573.

[19] D. Boneh, J. Bonneau, B. Bünz, and B. Fisch, "Verifiable delay functions," in *CRYPTO*, 2018, pp. 757–788.