Challenges of Proof-of-Useful-Work (PoUW)

Felix Hoffmann
Department of Computer Science and Mathematics
Goethe University
Frankfurt, Germany
felix.hoffmann@iri.uni-frankfurt.de

Abstract—Proof-of-Work is a popular blockchain consensus algorithm that is used in cryptocurrencies like Bitcoin in which hashing operations are repeated until the resulting hash has certain properties. This approach uses lots of computational power and energy for the sole purpose of securing the blockchain. In order to not waste energy on hashing operations that do not have any other purpose than enabling consensus between nodes and therefore securing the blockchain, Proof-of-Useful-Work is an alternative approach which aims to replace excessive usage of hash functions with tasks that bring additional real-world benefit, e.g. supporting scientific experiments that rely on computationally heavy simulations. In this publication theoretical PoUW concepts such as Coinami, CoinAI and the cryptocurrency Primecoin are analyzed with respects to how PoW properties can be retained while doing useful work.

Index Terms—blockchain, consensus algorithm, proof-of-work, proof-of-useful-work

I. INTRODUCTION

Traditional proof-of-work cryptocurrencies have been widely criticized for using up lots of energy in order to run and secure the underlying blockchain. In the past few years, there has been done research with the goal of replacing repeated hash operations with useful work. Notable projects such as Primecoin [1], Coinami [2] and CoinAI [3] use search for certain kinds of prime number chains, multiple sequence alignment of protein sequences or training of deep learning models as useful work consensus algorithms.

II. PROOF-OF-USEFUL-WORK (PoUW)

This section consists of two parts: In the first part, existing PoUW approaches and ideas are briefly introduced. In the second part, they are analyzed with regards to how the properties of hash-based PoW consensus algorithms are retained and which issues might occur. Even though exotic consensus algorithm classes like Proof-of-Storage can be considered useful, the focus in this publication is on computationally-heavy PoUW which shares lots of similarities with hash-based PoW.

A. Primecoin

Primecoin is a PoUW cryptocurrency that was launched in 2013 by Sunny King. [1] Its PoUW consists of finding certain types of prime number chains, so-called Cunningham and bi-twin chains. Cunningham chains are a series of prime numbers that nearly double each time. In mathematical terms, a prime chain of length \( n \in \mathbb{N} \) must fulfill

\[
p_{i+1} = 2p_i + 1 \quad (1)
\]
to be considered a first order chain or

\[
p_{i+1} = 2p_i - 1 \quad (2)
\]
to be considered a second order chain for all \( 1 \leq i < n \). For instance, \( \{41, 83, 167\} \) is a first order chain of length \( n = 3 \) and \( \{7, 13\} \) is a second order chain of length \( n = 2 \).

In addition to Cunningham chains, the third type of chain that Primecoin allows as proof-of-work are bi-twin chains. These are prime chains that consist of a strict combination of first and second order Cunningham primes. The mathematical definition of a bi-twin chain of length \( k + 1 \) is the sequence \( \{n-1, n+1, 2n-1, 2n+1, 2^2n-1, 2^2n+1, \ldots, 2^kn-1, 2^kn+1\} \). For instance, choosing \( n = 6 \) leads to \( \{5, 7, 11, 13\} \) which is a bi-twin chain of length 2 that consists of 4 prime numbers.

As of writing this publication, a Primecoin is traded for about $0.04 and the currency’s total market capitalization is around $1.7 million. [5] The success of Primecoin can be seen as evidence that PoUW is a viable concept with real-world applications.

B. Coinami

In 2016, a theoretical proposal of a mediator interface for a volunteer grid similar to BOINC middleware that can be connected to a cryptocurrency was published and named Coinami. [2] The PoUW of Coinami is built on DNA sequence alignment (HTS read mapping in particular) and aims to generate and analyze huge datasets of disease signatures which can help us to gain a better understanding of diseases such as different cancer variants.

The authors of Coinami describe their approach as a three-level multi-centric system which consists of...
a root authority, sub-authorities and miners. Miners
download problem sets from sub-authorities, map HTS
reads to a reference genome and send the results back
to sub-authorities for verification. Sub-authorities are
certified by the root authority. [2]

As a result, this approach can be seen as a hybrid of
Proof-of-Authority (PoA) and Proof-of-Useful-Work
(PoUW) consensus algorithms. As of writing, while
Coinami does have a prototype implementation on
Github [6], there currently exists no cryptocurrency
that is connected to this academic proposal.

C. CoinAI

In 2019, a theoretical proposal of PoUW consensus
that is built on training and hyperparameter optimization
of deep learning models was published and named
CoinAI. [3]
The goal of CoinAI is to secure a blockchain-based
cryptocurrency with a consensus algorithm that both
secures the underlying blockchain while also producing
deep learning models that solve real-world problems.
The proposed proof-of-work consists of training a
model that passes a certain performance threshold in
order for it to be considered valid. In addition to the
training of deep learning models, the CoinAI proposal
features another financial incentive to participate
in the blockchain: Nodes can rent out available
hard drive storage to provide distributed storage for
the resulting deep learning models of the blockchain. [3]

Thus, CoinAI’s approach can be described as a
hybrid of Proof-of-Useful-Work (PoUW) and Proof-
of-Storage (PoS). As of writing, CoinAI remains an
academic proposal that has not yet been implemented
to secure a tradeable cryptocurrency.

D. Analysis of PoUW approaches

1) Non re-usability: To prevent future calculation
and re-usability of proofs-of-work, a given problem
must involve information or parameters that can not
reliably be guessed beforehand. All nodes must be able
to agree on how these parameters are to be adjusted
over time so that the problem sets are adjusted over time
and it can be decided whether a given proof-of-work
is valid for some time interval. A common approach
here is to involve the hash of the previous block as a
parameter as part of the next problem. However, since
this directly influences the result of the calculations,
it must be decided on a case-per-case basis whether
the resulting information can still be considered to be
useful.

If incorporating hashes into the calculations is not
possible, then another approach must be found to
bind the PoUW to a given period in time. Relying
on an external (as in information taken from outside
the blockchain) source that continuously publishes
new information over time is not desirable, since this
approach leads to a high degree of centralization which
not only opposes core principles of a decentralized
blockchain but which also has the potential to create
security issues and conflicts of interest, especially if the
underlying blockchain is connected to a cryptocurrency.

▷ Primecoin retains the property of block sensitivity
by requiring the origin of the prime chains to be
divisible by the hash of the previous block. In this case,
the resulting quotient is defined as a so-called PoW
certificate. [1] This guarantees that pre-calculation of
future blocks is not a viable strategy as long as there
is no scientific breakthrough in efficiently calculating
certain chains of large primes.

▷ The theoretical Coinami approach tries to evade re-
usability and pre-calculation problems by relying on an
authority approach, in which miners must request tasks
from (sub)-authority nodes. Since miners can not guess
which task they might be given next, pre-calculation
of future blocks is not feasible. Since sub-authorities
know which problems have already been given out,
re-usability is not an issue either. The main issue of this
solution can be seen as a high degree of centralization
which forces miners to trust any (sub)-authority.

▷ The CoinAI proposal concatenates information
such as previous block hash, a random number called
nonce and a list of pending transactions which then
is hashed. This hash result then is used to determine
the initial hyperparameter structure of a deep learning
architecture which must be trained until it satisfies
performance requirements. An issue that potentially
arises with this approach is that if the goal is to produce
useful deep learning models, then starting the training
with an inadequate initial hyperparameter configuration
affects the amount of training required to reach
acceptable model performance which can be seen as
wasted energy. Assuming that the space of all allowed
hyperparameter configurations is limited to prevent this
from happening, the next problem that might arise is
that new hash-to-hyperparameter-configuration mapping
collisions are bound to happen more frequently, which
in this case means that multiple hashes lead to the
same initial hyperparameter configuration which as a
result could make pre-calculation strategies feasible.

2) Adjustable hardness: Since miners might join or
leave the network of nodes at any time, the blockchain’s
total computational power fluctuates over time. In order
to provide regular block intervals which in the case of a

cryptocurrency is necessary to stabilize the transaction

throughput, there must be consensus between nodes

with respect to how the difficulty of problems is to

be adjusted over time. Hash-based PoW approaches

control the problem difficulty by dynamically adjusting

the amount of leading zeroes that the resulting hash

must have in order to be valid depending on the current hash rate of the network. Increasing the amount of

required leading zeroes by just one increases the
difficulty of the hash puzzle exponentially, which is

why softer variations of this approach can be used

(such as e.g. amount of leading digits smaller than
eight) to provide a more fine-grained control of the

problem difficulty.

For useful work approaches, it needs to be decided

on a case-per-case basis how the hardness of a
given problem can dynamically be adjusted without

jeopardizing usefulness of results.

In the context of Primecoin, two intuitive mechanics
to control problem difficulty come to mind: First of all,
the size of prime numbers that start a chain could
be increased over time. However, the prime number

theorem states that

\[ \lim_{x \to \infty} \frac{\pi(x)}{\frac{x}{\ln(x)}} = 1 \]  

with \( \pi(x) \) being the so-called prime-counting function.
The for our context useful interpretation of this

equation is that the prime density approaches zero,

which means that the proof-of-work difficulty over time

might become too high to sustain stable transaction

throughput long-term.

The second intuitive approach that comes to mind is to dynamically adjust the required length of valid

prime number chains to control the problem difficulty. This is the approach Primecoin takes: Given a

prime chain of some length, Primecoin dynamically

adjusts its Fermat primality test which results in a

relatively linear continuous difficulty function (as

opposed to the non-linear difficulty function of the first

approach) that is claimed to be accurate enough to

adjust the problem hardness appropriately over time. [1]

The Coinami authors have not yet defined how

the difficulty of the DNA sequence alignment problems

can be dynamically adjusted over time. The issue here

is that the network must rely on an external source for

HTS data and simply increasing the size of assignments

potentially leads to issues with resulting data size and

networking bottlenecks. An idea here is to let miners

solve multiple problems at once and then let authority

nodes randomly select one of these solutions and
discard the others. While this can be seen as a waste of

useful work it might be necessary sacrifice to control

problem difficulty without increasing data sizes.

A core principle of consensus algorithms in public blockchains is that they are used in order to provide nodes with a method that enables

them to form consensus about the current state of the blockchain without having to rely on trust. Hash functions are useful in this regard since the validity of a proposed (input, output) tuple can quickly be verified. As soon as hash-based approaches are discarded in favor of methods that perform useful work, it can become difficult to find a verification method that does not have to rely on a verification-by-replication approach in which the entire useful work process has to be repeated by many nodes. For a given problem there might or might not exist a probabilistic verification approach in which the likelihood of some proposed solution being valid can be estimated efficiently. Therefore, it needs to be decided on a case-per-case basis what is the best way to formulate a PoUW problem in such a way that verification of results can happen quickly and with reasonable amounts of computational effort.

In the case of Primecoin, probable primality of
prime chains is verified using a combination of both the Fermat and the Euler-Lagrange-Lifchitz test for prime numbers. These are proven mathematical methods that can be used to efficiently verify the primality of a given number with the downside that there exist so-called pseudoprimes that pass those prime tests but which are in fact not prime numbers. The authors of Primecoin have concluded that the probability of pseudoprimes occurring is low enough that this issue can be traded in favor of being able to provide a fast and efficient verification mechanism. [1]

▷ In the Coinami proposal, sub-root authorities collect results from miners and verify the validity of alignments using decoy reads that have been placed into the problem. These decoys are planned to make up around 5% of each problem and they can be pre-calculated by the sub-authorities. After verification, decoy data is removed from the results. The main challenge here is to place decoy data in such a way that miners are not able to spot these segments in their assignments. If a sub-authority has validated a miners solution, then the data is signed and sent back so that it can be added to the blockchain.

▷ In CoinAI resulting deep learning models are considered to be valid proofs-of-work only if they pass the current performance threshold. The authors provide no concrete plans about whether a centralized entity is responsible for verification or if every miner has to verify all submitted models by other nodes. Potential issues that might occur in either case have already been presented in the adjustable hardness section of this publication. II-D2

A common approach to validate the performance of a deep learning model is to use two separate datasets, one containing training data used for training the model and the second dataset being the validation/test dataset. CoinAI gives no specifics on how nodes acquire required training datasets which potentially poses a challenge in overcoming issues such as networking bottlenecks due to large datasets that need to be downloaded. The current state-of-the-art in training of deep learning models boils down to the fact that you need more and more training data to improve your model over time, since hyperparameter tuning of a model that was trained on a small dataset alone rarely results in a robust model than can reliably solve non-trivial problems. As a result, the training dataset would have to be extended over time which raises further questions about who provides this data, how this affects centralization and who is willing to sacrifice computational power and network bandwidth to test the performance of all submitted models. Even if all of these potential issues were to be resolved, assuming the same model is trained over many blocks one could argue that as soon as better performing models for a given task are discovered all previously published models lose their usefulness since they perform worse than the newer models. This raises the question if such an approach can be considered to be useful work in the first place. If, however, completely different deep learning models are to be trained at regular block intervals, potential problems of continuously broadcasting new training data sets and generating robust models performance might become overwhelming.

4) Parallelizability: In order to enable the efficient usage of multi-core CPUs, GPUs and facilitate the existence of mining pools, a PoUW consensus algorithm preferable should be of embarrassingly parallel nature. An intuitive example of such a problem is any form of processing or generation of unrelated data, like it is done in e.g. brute-force searches. There are many non hash-based approaches that fulfill this property: For instance, Monte Carlo event generation and reconstruction in particle physics, pattern matching over DNA sequences in bioinformatics and hyperparameter tuning in deep learning can all be considered to be embarrassingly parallel problems.

▷ In Primecoin the search for prime chains can trivially be implemented in a parallelizable way.

▷ Pattern matching over DNA sequences in bioinformatics like proposed in Coinami is of embarrassingly parallel nature.

▷ Training deep learning models and hyperparameter tuning like proposed in CoinAI is an embarrassingly parallel problem.

All in all, it can be concluded that retaining the parallelizability property is not an issue for PoUW approaches.

III. CONCLUSION

This publication has provided an analysis of which measures were taken by existing PoUW approaches such as Primecoin, Coinami and CoinAI to retain hash-based PoW properties while rewarding useful work. It was concluded that domain-specific knowledge is required to make PoUW consensus possible and that implementation details must be decided on a case-by-case basis using domain knowledge from that area of research.
The main weakness that all presented PoUW approaches have in common is the verification of results. While the author of Primecoin was able to find an elegant probabilistic solution of this problem, theoretical publications like Coinami and CoinAI had to make both efficiency and decentralization sacrifices to prevent potential problems.

A common issue with designing new PoUW consensus approaches is that the size of resulting data can be significant compared to hash-based approaches which leads to situations in which data must either be stored externally or on-chain which negatively affects not only storage requirements of full nodes but also sync times of new nodes which effectively raises the entry barriers of participating in the blockchain.

All in all, problems of mathematical nature seem to be best suited for PoUW. These problems have the advantage that a large repertoire of probabilistic verification methods already exists for a wide range of problems, which in addition to a generally asymmetrical ratio of computational effort and size of resulting output make this class of problems potential suitable for making PoUW consensus mainstream.

It remains to be seen whether the concept of Proof-of-Work itself will survive the surge of alternative blockchain consensus algorithms like Proof-of-Stake which do not require notable amounts of computational effort to efficiently form consensus and therefore secure the underlying blockchain.

REFERENCES

[1] S. King. Primecoin: Cryptocurrency with prime number proof-of-work. https://www.primecoin.io/primecoin-paper.pdf, 2013.
[2] A. M. Ileri, H. I. Ozercan, A. Gundogdu, A. K. Senol, M. Y. Oezkaya, and C. Alkan. Coinami: A cryptocurrency with dna sequence alignment as proof-of-work. CoRR, abs/1602.03031, 2016.
[3] A. Baldominos and Y. Saez. Coin.ai: A proof-of-useful-work scheme for blockchain-based distributed deep learning. Entropy, 21(8):723, 2019.
[4] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system. https://www.nakamotoinstitute.org/literature/bitcoin/, 2008.
[5] Coinmarketcap. Primecoin. https://www.coinmarketcap.com/currencies/primecoin/, 2022.
[6] Github. Coinami prototype. https://www.github.com/coinami/coinami-pro, 2016.