Fruitchains: A Fair Blockchain?

Kyrylo Voronchenko
Supervised by Michal Zajac

June 4, 2018

Abstract

This report is an overview of the paper [PS16] in which authors introduced a new protocol called `FruitChain` that has fair reward mechanism, removes transaction fees instability and eliminates the need of mining pools. They shown that proposed protocol can be run as an instance of Nakamoto's blockchain protocol.

Contents

1 Introduction

The word blockchain stems from its technical structure - a chain of blocks. Each block in the blockchain is connected to the previous one with a cryptographic hash along with a timestamp. A block is a data structure that can store a list of records. In Bitcoin, when miners try to compute the block, they pick all transactions they want and include them into the block so they can form a tree of transaction later hashed into a merkle root and referenced into a block's header.

Miners are incentivized to solve computaional puzzles by receiving block rewards for every block together with transactions fee from transactions they included into the mined block. Nakamoto's Bitcoin blockchain protocol reward mechanism suffers from several attacks:

1. *Selfish-mining attack.*

   In selfish-mining attack, if an adversary controls the network delivery, he may get close to half of all the rewards. Adversary withholds his mined block and wait until honest party will solve computational puzzle and mine a new block. As an adversary controls network delivery, he can deny incoming honest party block and replace it with his own already mined block.

2. *Transaction fees exacerbate instability.*

   In Nakamoto's Bitcoin, the block reward is halves every 210,000 blocks. Due to this property, miners are expected to obtain most of the reward from transaction fees. In the paper, authors mentioned that a recent work by Carlsten at al.[CKWN16] show that if a block contains transactions with large fees, miners will be incentivized to create a "fork" and attempt to confirm transactions themselves[PS16]

3. *Mining pool harm decentralization.*

   Nowadays, the block difficulty setted in such way that one block needs to be mined in approximatelly 10 minutes[GKL14]. It is impossible to individual miner for mine a new block during this period of time. Today, a solo miner wourld take 2 to 5 years to obtain its first reward. [sol]. That is why participants of the network started to collaborate with each other and share their mining power. Mining pool is a group of people that share their computational resources in order to mine a block and after poll operator splits the block reward proportional to the hashing power they used. Mining pools remove the main feature of a blockchain - decentralization. For instance, BTC.com mining pool has 28.4% of all the Bitcoin mining power[1]. Also, most of the mining pools participants mining power are located in China and if goverment will try to ban a mining pool, it most probably will do it.

Fruitchain protocol removes these three properties that were stated above. In fruitchain, we have 2 different data structures - fruits and blocks. Fruits consists of transaction data and blocks are consists of fruit. Each data structure has its own proof-of-work that is need to be solved for mining. The difficulties of these proof-of-work are different and selected in that way, that even individual miner is able to mine a fruit of a block.

2 Protocols Overview

2.1 Nakamoto's Blockchain Protocol

Satoshi Nakamoto proposed a decentalized and distributed solution for maintaining for chain of blocks of records - called a blockchain. Each block of a blochchain consist of four main parts:

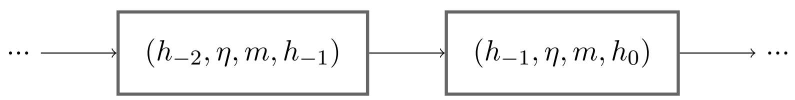

Figure 1: Bitcoin Blockchain Structure

, where:

1. h_{-1} - pointer to the hash of the previous block
2. η - nonce(proof of work derived from the pair (h_{-1}, m).
3. h - pointer to the current record
4. m - a set of records (transactions).

Bitcoin application blockchain protocol is parametrized by a mining harness difficulty parameter p and proof-of-work is deemed valid if η is a string such that $H(h_{-1}, \eta, m) < D_p$ where H is a hash function is set that an inpit satisfies the relation is less than p.[**?**]. Basically, this means that the string projects into whole range of hash function values and if this string satisfies

[1] https://blockchain.info/pools

an inequation, this string is a correct input. Parameter p acts as a value "resized" the range of the values that may satisfy the relation.

Each participant of the protocol, may attempt to receive the set of messages m from other participants, pick random η, check whether it satisfies the the relation above and extend the blockchain by sending his mined block over network.

2.2 The FruitChain Protocol

FruitChain differs from Nakamoto's blockchain protocol. Compare to Bitcoin blockchain, instead of putting a set of messages m into the block, authors come up with idea to put them into the "fruits" denoted as f.

2.2.1 Fruit structure

Fruits structure are similar to Bitcoin block structure but fruit requires additional requirements to be 'hanged' (mined):

1. Fruits requires solving additional proof-of-work with a different handness parameter p_f.

2. Fruit should store a pointer to a block that is not far away from the block containing it. Authors proposed to use additional parameter R that specifies how far back a fruit allows to be.

A valid fruit has the following structure:

$$\boxed{(h_{-1}, h', \eta, digest, m, h)}$$

Figure 2: Fruit Structure

, where:

1. h_{-1} - pointer to the hash of the previous block.

2. h' - pointer to a block from which the fruit f hangs

3. η - nonce, fruit proof-of-work solution. The mining hardness parameter parameter is p_f which is different from block p.

4. *digest* - digest of fruit set F that contains fruit f. Instead of set of records (transactions), block contains the fruit set.

5. m - message (records) that fruit contains.

6. h - hash of the fruit.

Definition 2.1 *Valid fruit (from [PS16])*
A fruit denoted as $f = (h_1, h', \eta, \boldsymbol{digest}, \boldsymbol{m}, h)$ *is valid iff:*

1. $H(h_{-1}, h', \eta, \boldsymbol{digest}, \boldsymbol{m}) = h$

2. $[h]_{-\kappa} < D_{p_f}$ *where* $[h]_{-\kappa}$ *denotes the last* κ *bits of* h*. We say that* F *is a valid fruit-set if either* $F = \emptyset$ *or* F *is a set of valid fruits.*

2.2.2 Block structure

A block structure in a `FruitChain` protocol is similar to a fruit structure except it also requires the fruitset F.

$$\boxed{((h_{-1}, h', \eta, \mathbf{digest}, \mathbf{m}, h), F)}$$

Figure 3: `FruitChain` Block Structure

, where:

1. h_{-1} - a pointer to a previous block in a chain.

2. h' - is an artifact of the fruit mining and block mining piggybacked.

3. η - nonce denoting proof-of-work solution for a block. Difficulty parameter p denotes the mining hardness difficulty for block proof-of-work, so it is different from p_f that denotes the fruit mining hardness difficulty.

4. *digest* - digest (hash) of a fruit set F that will be included to this block.

5. m - a record, artifact of the two mining processes.

6. h - a pointer to a block, hash of all the previous values

7. F - fruit set that needs to be included into the block.

Fruit set F has the following structure:

$$\boxed{\boxed{f_0} \qquad \boxed{f_1} \qquad \boxed{f_2} \qquad \boxed{f_3}}$$

Figure 4: Fruit Set Structure

Note that fruits are not linked to each other. Each fruit has a reference to a block that is not too far away from the newest block in a blockchain.

Definition 2.2 *Valid block (from [PS16])*
A block denoted as $b = ((h_{-1}, h', \eta, \boldsymbol{digest}, \boldsymbol{m}, h), F)$ *is valid iff:*

1. $\boldsymbol{digest} = d(F)$, *where d is a collision-resistant hash function.*
2. *F is a valid fruit-set.*
3. $H(h_{-1}, h', \eta, d(F), m) = h$
4. $[h]_{:\kappa} < D_{p_1}$ *where* $[h]_{:\kappa}$ *denotes the first κ bits of h.*

2.2.3 Blockchain Structure

`FruitChain` blockchain has the following structure:

$$\cdots \longrightarrow \boxed{((h_{-2}, h', \eta, \mathbf{digest}, \mathbf{m}, h_{-1}), F_0)} \longrightarrow \boxed{((h_{-1}, h', \eta, \mathbf{digest}, \mathbf{m}, h_0), F_1)} \longrightarrow \cdots$$

Figure 5: `Fruitchain` Blockchain Structure

Definition 2.3 *Valid blockchin (from [PS16])*
We say that a chain is valid iff:

1. *chain*$[0]$ = *genesis where genesis* := $((0; 0; 0; \perp; H(0; 0; 0; 0, \perp)), \emptyset)$ *is the "genesis" block.*
2. *for all* $i \in [l]$, *chain*$[i].h_{-1}$ = *chain*$[i-1].h$, *i.e., each block refers to the previous block's pointer.*
3. *for all* $i \in [l]$, *all* $f \in$ *chain*$[i].F$, *there exists some* $j \geq i - R\kappa$ *such that the pointer of f is chain*$[j].h$.

As we can see, the first two properties are similar to Nakamoto's valid blockchain definition that differs only in block structure. Third property states that fruit set F needs to be not far back from the block to which fruits from this set point to.

2.2.4 Fairness in `FruitChain` blockchain protocol

In the [PS16], authors introduces a notion of fairness for blockchain protocols.

Definition 2.4 *Fairness (from [PS16]) A blockchain protocol is fair if honest players that wiled ϕ fraction of the computational resources will reap al least ϕ fraction of the blocks in any sufficiently long window of the chain.*

In their work, they proved that `FruitChain` blockchain is fair and has consistency and liveness properties as Nakamoto's one. If the protocol is fair, adversary is not able to gain "much" more that its fair share of a block rewards and transaction fee. That is why the selfish-mining problem disincentivized.

2.2.5 Transaction fees exacerbate instability

In [PS16], authors suggested a method for spreading out the transaction fees of a block over the miners of a sequence of blocks preceeding it.

2.2.6 Disincentivizing Pooled Mining

As a solution for mining pools centralization problem, authors of [PS16] proposed Fruichain protocol that is parametrized by two mining hardness parameters - p for a block and p_f for a fruit. These parameters are independent of each other, so p can be set appropriately to ensure consistency of a whole blockchain and p_f can be set much larger. For instance, the difficulty p_f may be equal to a partial proof-of-work that "solo-miner" can solve without being in a mining pool. In [PS16], authors calculated that if they allocate space for 1000 fruits in a block, where each block is 80 bytes, this would occupy roughly 8% of a 1MB block and this will allow a solo miner to get his first rewards 1000x faster compare to a Bitcoin.

3 Summary

Blockchain technology is promising technology that yet has to find its new opportunities and new application. In the [PS16] paper, authors proposed a new blockchain protocol that is fair, deincentivizes mining pools and give an opportunity to earn a reward for a solo miners. I hope that we will see the implementation of this protocol in the nearest future.

4 References

References

[CKWN16] Miles Carlsten, Harry Kalodner, S. Matthew Weinberg, and Arvind Narayanan. On the instability of bitcoin without the block reward, 2016.

[GKL14] Juan Garay, Aggelos Kiayias, and Nikos Leonardos. The bitcoin backbone protocol: Analysis and applications. Cryptology ePrint Archive, Report 2014/765, 2014. `https://eprint.iacr.org/2014/765`.

[Nak] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, http://bitcoin.org/bitcoin.pdf.

[PS16] Rafael Pass and Elaine Shi. Fruitchains: A fair blockchain. Cryptology ePrint Archive, Report 2016/916, 2016. `https://eprint.iacr.org/2016/916`.

[sol] `https://www.coinwarz.com/calculators/bitcoin-mining-calculator`.