Proof-of-Stake Blockchain Security

Michael Färber

22 November 2017

Quiz

Figure 1: What am I?

Literature

Ouroboros

- Aggelos Kiayias, Alexander Russell, Bernardo David, Roman Oliynykov: Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol. CRYPTO (1) 2017: 357-388
- First proof-of-stake blockchain with formal security analysis

This got my attention ...

- 2017W703049 VO Prinzipien von Blockchain-Systemen (held by Rainer Böhme)
- Cardano - next generation blockchain platform written in Haskell (Haskell Reddit thread)

Byzantine Generals' Problem[1]

Problem

- Group of generals discuss over distance whether to attack
- Some generals may be traitors and may send conflicting messages to others, trying to create conflicting actions (50% attack, 50% retreat)

Result

Consent can be only guaranteed when $< \frac{1}{3}$ of generals are traitors

[1] Leslie Lamport, Robert E. Shostak, Marshall C. Pease: The Byzantine Generals Problem. ACM Trans. Program. Lang. Syst. 4(3): 382-401 (1982)

Byzantine Generals' Problem for Blockchain

Re: Bitcoin P2P e-cash paper

Satoshi Nakamoto | Thu, 13 Nov 2008 19:34:25 -0800

James A. Donald wrote:
> It is not sufficient that everyone knows X. We also
> need everyone to know that everyone knows X, and that
> everyone knows that everyone knows that everyone knows X
> - which, as in the Byzantine Generals problem, is the
> classic hard problem of distributed data processing.

The proof-of-work chain is a solution to the Byzantine Generals' Problem. I'll try to rephrase it in that context.

A number of Byzantine Generals each have a computer and want to attack the King's wi-fi by brute forcing the password, which they've learned is a certain number of characters in length. Once they stimulate the network to generate a packet, they must crack the password within a limited time to break in and erase the logs, otherwise they will be discovered and get in trouble. They only have enough CPU power to crack it fast enough if a majority of them attack at the same time.

Figure 2: Proof-of-Work chain solves the Byzantine Generals' Problem

Blockchain

- List of records (*blocks*) shared across network → decentralised
- Every block contains the hash of its predecessor → immutable
- Current most significant representative: Bitcoin

Blockchain formation

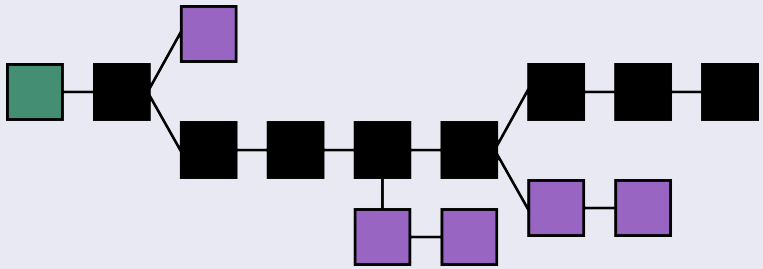

Figure 3: Blockchain formation.

Who generates Blocks?

Leader Selection

Every round of the protocol, a chosen leader can create a block, incorporating data (e.g. transactions) from other participants.

Proof-of-Work (Example: Bitcoin)

- Leader is entity that finds nonce such that hash of (previous block hash$\|$data$\|$nonce) is below a threshold value
- Cons: work costs large amounts of energy

Proof-of-Stake

- Leader is selected randomly from stakeholders, weighted by stake
- Cons: stakeholders need to participate in protocol

Ouroboros: Committees select Leaders

Committee

- Time is divided into *epochs* (fixed number of rounds)
- Every epoch, some stakeholders are randomly selected to form a *committee*
- Committee randomly selects leaders
- Problem: Committee members have to be online during epoch

Solution: Stake Delegation

- Stakeholders can delegate participation in committees to *delegates*
- Delegates must prove that the aggregate stake of their voters is above a certain threshold (to ensure protocol performance)

Excursion: Delegative (a.k.a. Liquid) Democracy

Direct Democracy

- Everybody can vote on all matters
- Pure direct democracy requires high participation effort

Representative Democracy

- Voters elect representatives that have equal influence
- High entry barrier to become representative (candidate)

Delegative Democracy

- Voters temporarily delegate power to a delegate that has influence proportional to its voter support
- In-between direct and representative democracy

Restrictions on Protocol

Adversary

- Adversary can corrupt other entities with fixed minimal time delay[a]
- Cumulated stake of adversary plus corrupted entities is less than 50%

[a] Time delay restriction is relaxed in successor paper

Honest parties

- Every honest party can be offline only for a fixed maximum time
- Very conservative restriction; in practice protocol tolerates longer offline times

Plan of (Proof) Attack

Refinement of proof for increasingly complex setting

1. Static stake
2. Dynamic stake with global beacon to seed leadership election
3. Dynamic stake without beacon
4. Input endorsers, stakeholder delegates, anonymous communication (not discussed here)

Forkable Strings

Motivation: Double Spending Attack

1. Adversary creates diverging views on blockchain for honest parties
2. Adversary pays the same money two times

Question

Under which conditions can adversary make honest parties adopt diverging views on the blockchain?

Characteristic String

The characteristic string $w \in \{0,1\}^n$ is

- 0 in case a honest party was elected leader and
- 1 if an adversary-controlled (corrupt) party was elected leader

Forks and Forkability

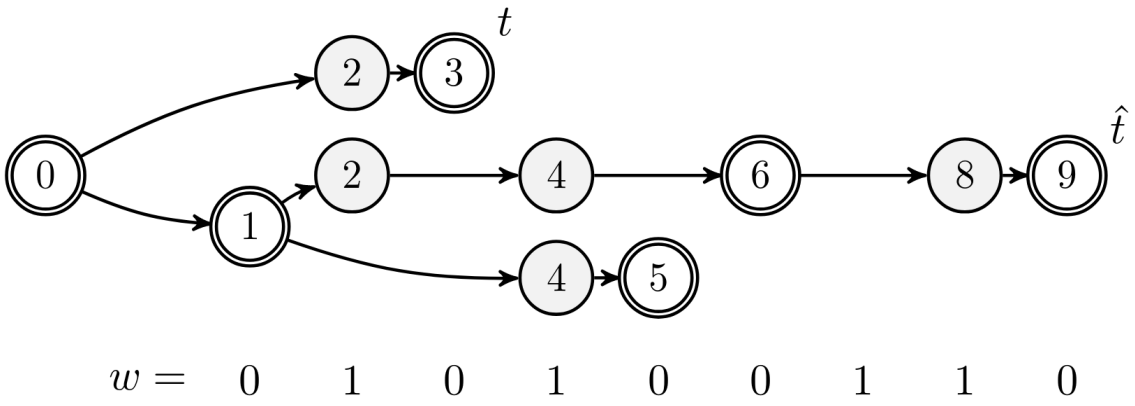

Figure 4: Fork for the string $w = 010100110$

A characteristic string w is forkable if there is a fork corresponding to w with two paths, such that the length of both paths is the height of the fork.

Theorem

Let $\epsilon \in (0, 1)$ and let w be a string drawn from $\{0, 1\}^n$ by independently assigning each $w_i = 1$ with probability $(1 - \epsilon)/2$. Then

$$P(w \text{ is forkable}) = 2^{-\Omega(\sqrt{n})}.$$

Protocol Properties

- Common Prefix (CP) with $k \in \mathbb{N}$: All honest parties have the same blocks up to the most recent k blocks.
- Chain Quality (CQ) with $\mu \in (0, 1]$ and $\ell \in \mathbb{N}$: Every section of the chain of length ℓ of a honest party has a ratio of blocks from the adversary of at most $1 - \mu$.
- Chain Growth (CG) with $\tau \in (0, 1]$ and $s \in \mathbb{N}$: For any two chains of honest parties where the longer chain is at least s time slots ahead, the difference of the chain lengths is at least $\tau \cdot s$.

Property	Maximal violation probability
CP	$\exp(-\Omega(\sqrt{k}) + \ln R)$
CQ	$\exp(-\Omega(\epsilon^2 \alpha \ell) + \ln R)$
CG	$\exp(-\Omega(\epsilon^2 s) + \ln R)$

R is epoch duration, ϵ is stake advantage of honest parties over adversary and α is rate of adversarial parties. We set $\tau = 1 - \alpha$.

Does it make sense to be honest?

Rationality of honesty

- So far: Majority of honest players execute protocol faithfully
- However, stakeholders are not necessarily honest, but *rational*
- Is it rational to be honest, i.e. do you profit most being honest?

Nash equilibrium

Strategy is Nash equilibrium if any single player diverging from its current strategy does not gain advantage

Theorem

The honest strategy in the Ouroboros protocol is a Nash equilibrium

Transaction Confirmation Time

Table 2: Transaction confirmation times in minutes to exclude double spending attacks with 99.9% certainty.

Adversary	BTC	OB
0.10	50	5
0.15	80	8
0.20	110	12
0.25	150	18
0.30	240	31
0.35	410	60
0.40	890	148
0.45	3400	663

Summary

- Ouroboros: new proof-of stake blockchain protocol
- Stake delegation relaxes requirement of nodes to be online
- Security depends on several assumptions, e.g. maximal adversary power
- Maximal violation probability of important properties shown
- Being honest is rational – if the system encourages honesty :)
- Transaction confirmation time is lower than that of BTC

Summary

- Ouroboros: new proof-of stake blockchain protocol
- Stake delegation relaxes requirement of nodes to be online
- Security depends on several assumptions, e.g. maximal adversary power
- Maximal violation probability of important properties shown
- Being honest is rational – if the system encourages honesty :)
- Transaction confirmation time is lower than that of BTC

How about a formally verifying the proof? :)